

C⁴I BRANCH
BIBLIOGRAPHY / ADDITIONAL READINGS
FOR
SOFTWARE METRICS GUIDEBOOK

OCTOBER 1995

Submitted
by the
Applied Research Laboratory
of the
Pennsylvania State University (PSU)

James A. Hager - Research Associate
Richard Chen - PSU EE Graduate Student

INDEX

Cocomo	1
Communication/Interaction Measures	1
Concurrency Measurement	1
Control Flow Measures	1
Correlations	2
Cost/Effort/Size Estimation	3
Data (Flow) Measures	6
Evaluation	6
Evaluation/Certification Measures	7
Factor Analysis	8
Formal Specification	9
Function Points	9
Functional Programming	11
Halsteads Software Science	11
Hybrid Measures	13
Hypergraph Measures	13
Hypertext Metrics	13
Information Measures	13
Logical Programming	14
Maintenance Metrics	14
Management	16
Management Bibliography	17
Mccabes Cyclomatic Number	26
Measurement Foundation	27
Measurement Frameworks	29
Metric Analysis	32
Metrics And Formal Grammars	34
Metrics Bibliography	34
Metrics Validation	40
Modifiability Measures	41
Modularization Measurement	42
Network Measurement	43
Object-Oriented Design Measures	44
Object-Oriented Programming	44
Object-Oriented Software Engineering Bibliography	46

Object-Oriented Testing	103
Overview	103
Performance Measurement	104
Portability Measures	104
Process Analysis And Management	104
Program Analysis	105
Programmers Productivity Measures	106
Pseudocode Measures	107
Real-Time Programming Measurement	107
Regression Models	107
Reliability Measurement	107
Requirement Analysis	109
Reusability Measures	109
Reuse Bibliography	110
Review Analysis Measures	125
Software Factory	125
Software Management	126
Software Process Improvement	128
Software Quality	131
Software Quality Assurance	136
Software Systems/Tree Structure	137
Source Code Measures	138
Standards	139
Testing Bibliography	140
Test Coverage	151
Test Metrics	152
Text Analysis	154
Visual Programming	154

COCOMO

1. Boehm, B.W.: Les facteurs du cout du logiciel. T.S.I., Paris, 1(1982)1, pp. 5-24.
2. Boehm, B.W.: Software Engineering Economics. IEEE Transactions on Software Engineering, 10(1984)1, pp. 4-21.
3. Boehm, B.W.; Papaccio, P.N.: Understanding and Controlling Software Costs. IEEE Transactions on Software Engineering, 14(1988)10, pp. 1462-1477.
4. Kitchenham, B.A.; Taylor, N.R.: Software Project Development Cost Estimation. The Journal of Systems and Software, 5 (1985), pp. 267-278.
5. Marwane, R.; Mili, A.: Building tailor-made software cost model: Intermediate TUCOMO. Information and Software Technology, 33(1991)3, pp. 232-238.
6. Miyazaki, Y.; Mori, K.: COCOMO Evaluation and Tailoring. Proceedings of the 8th Conference on Software Engineering, August 28-30, London, 1985, pp. 292-299.
7. Miyazaki, Y.; Murakami, N.: Software Metrics Using Deviation Value. Proceedings of the 9th International Conference on Software Engineering, March 30 - April 2, Monterey, 1987, pp. 83-91.

COMMUNICATION/INTERACTION MEASURES

1. Beane, J.; Giddings, N.; Silverman, J.: Quantifying Software Designs. Proceedings of the 7th Conference on Software Engineering, Orlando, Florida, March 1984, pp. 314-322.
2. Gannon, J.D.; Katz, E.E.; Basili, V.R.: Metrics for ADA Packages: An Initial Study. Comm. of the ACM, 29(1986)7, pp. 616-623.
3. Levine, D.L.; Taylor, R.N.: Metric-Driven Reengineering for Static Concurrency Analysis. Proc. of the International Symposium on Software Testing and Analysis (ISSTA), June 28-30, Cambridge, MA, 1993, pp. 40-50.
4. Hall, N.R.; Preiser, S.: Dynamic Complexity Measures for Software Design. IEEE Computer, (1983), pp.~57-66.
5. Shatz, S.M.: Towards Complexity Metrics for Ada Tasking. IEEE Transactions on Software Engineering, 14(1988)8, pp. 1122-1127.

CONCURRENCY MEASUREMENT

1. Yau, S.S.; Chen, F.: An Approach to Concurrent Control Flow Checking. IEEE Transactions on Software Engineering, 6(1980)2, pp. 126-137.

CONTROL FLOW MEASURES

1. Baker, A.L.; Zweben, S.H.: A Comparison of Measures of Control Flow Complexity. IEEE Transactions on Software Engineering, 6(1980)6, pp.~506-512.
2. Binder, L.H.; Poore, J.H.: Field Experiments With Local Software Quality Metrics. Software -- Practice and Experience, 20(1990)7, pp. 631-647.

3. Cantona, G.; Cimitile, A.; Sansone, L.: Complexity in Program Schemes: The Characteristic Polynomial. SIGPLAN Notices, 18(1983)3, pp. 22-30.
4. Chapin, N.; Denniston, S.P.: Characteristics of a Structured Program. SIGPLAN Notices, 13(1978)5, pp. 36-45.
5. Chen, E.T.: Program Complexity and Programmer Productivity. IEEE Transactions on Software Engineering, 4(1978)3, pp. 187-194.
6. Evangelist, W.M.: Software Copmplexity Metric Sensitivity to Program Structuring Rules. The Journal of Systems and Software, 3(1983), pp. 231-243.
7. Fenton, N.E.; Whitty, R.W.: Program Structure: Some New Characterisations. Journal of Computer and System Science, 43(1991)3, pp. 467-483.
8. Fitzsimmons, A.; Love, T.: A Review And Evaluation Of Software Science. Tutorial -- Programming Productivity: Issues for the Eighties, IEEE Computer Society, ISBN 0-8186-0681-9, 1986, pp. 45-60.
9. Forgacs, I.: Using Dependence Graphs for Software Quality. The Software QA, 1(1994)1, Baverton, OR, pp. 18-25.
10. Howatt, J.W.; Baker, A.L.: Rigorous Definition and Analysis of Program Complexity Measures: An Example Using Nesting. The Journal of Systems and Software, 10(1989)2, pp. 139-150.
11. Hecht, M.S.: Flow Analysis of Computer Programs. Elsevier Publisher, New York, 1977.
12. Jayaprakash, S.; Lakeshman, K.S.; Sinha, P.K.: MEBOW: A Comprehensive Measure of Control Flow Complexity. Proceedings of the COMPSAC, Tokyo, October 7-9, 1987, pp. 238-244.
13. Mennert, A.: Measuring Control Flow Complexity for Software Development. Technical Report, Siemens, Princeton, New Jersey, 1991.
14. Piwowarski, P.: A Nesting Level Complexity Measure. SIGPLAN Notices, 17(1982)9, pp. 44-50.
15. Prather, R.E.: Hierarchical Software Metrics. Tutorial on the Second International Conference on Software Quality, Research Triangle Park, NC, October 1992.
16. Robillard, P.N.; Simoneau, M.: A New Control Flow Representation. Proceedings of the 15th COMPSAC, Tokyo, September 11-13, 1991, pp. 225-230.
17. Robillard, P.N.; Simoneau, M.: Iconic Control Graph Representation. Research Report, 18 p., Ecole Polytechnique Montreal, 1992.
18. Woodward, M.R.; Hennell, M.A.; Hedley, D.: A Measure of Control Flow Complexity in Program Text. IEEE Transactions on Software Engineering, 5(1979)1, pp. 45-50.

CORRELATIONS

1. Basili, V.R.; Selby, R.W.; Phillips, T.: Metric Analysis and Data Validation Across Fortran Projects. IEEE Transactions on Software Engineering, 9(1983)6, pp. 652-663.

2. Courtney, R.E.; Gustafson, D.A.: Shotgun correlations in software measures. *Software Engineering Journal*, January 1993, pp. 5-13.
3. Crawford, S.G.; McIntosh, A.A.; Pregibon, D.: An Analysis of Static Metrics and Faults in C Software. *The Journal of Systems and Software*, 5 (1985), pp. 37-45.
4. Grimm, E.: Correlation about software complexity measures (german). Master's Thesis, TU Berlin/TU Magdeburg, 1991.
5. Kitchenham, B.A.; Linkman, S.J.: Design metrics in practice. *Information and Software Technology*, 32(1990)4, pp. 304-310.
6. Li, H.F.; Cheung, W.K.: An Empirical Study of Software Metrics. *IEEE Transactions on Software Engineering*, 13(1987)6, pp. 697-708.
7. Lipow, M.: Number of Faults per Line Code. *IEEE Transactions on Software Engineering*, 5(1979)2, pp. 76-79.
8. Myrvold, A.: Data Analysis for Software Metrics. *The Journal of Systems and Software*, 12 (1990), pp. 271-274.
9. Pickard, L.M.: Analysis of Software Metrics. in: Kitchenham; Littlewood: *Measurement for Software Control and Assurance*. Elsevier Science Publisher Ltd, 1989, pp. 155-18.
10. Shen, V.Y.; Yu, T.; Thebaut, S.M.; Paulsen, L.R.: Identifying Error-Prone Software -- An Empirical Study. *IEEE Transactions on Software Engineering*, 11(1985)4, pp. 317-324.
11. Woodfield, S.N.; Shen, V.Y.; Dunsmore, H.E.: A Study of Several Metrics for Programming Effort. *The Journal of Systems and Software*, (1981)2, pp. 97-103.

COST/EFFORT/SIZE ESTIMATION

1. Abdel-Hamid, T.K.: Adapting, Correcting, and Perfecting Software Estimates: A Maintenance Metaphor. *IEEE Computer*, March 1993, pp. 20-29.
2. Arifoglu, A.: A Methodology for Software Cost Estimation. *Software Engineering Notes*, 18(1993)2, pp. 96-105.
3. Bailey, J.W.; Basili, V.R.: A Meta-Model for Software Development Resource Expenditures. *Proceedings of the 5th International Conference on Software Engineering*, March 9-12, San Diego, California, 1981, pp. 107-116.
4. Banker, R.D.; Kemerer, C.F.: Scale Economics in New Software Development. *IEEE Transactions on Software Engineering*, 15(1989)10, pp. 1199-1205.
5. Basse, B.: Investigations to the applicability of PC-HOST connections (german). Master's Thesis, TU Magdeburg, 1992.
6. Carat, C.-M.: Usability Engineering in Dollars and Cents. *IEEE Computer*, May 1993, pp. 88-89.
7. Cost Estimation Measures. *Tutorial of the SEMA-GROUP -- Knowledge applied*, England, 1991.

8. Costing the coding. Application Brief, Logic Programming Associates (LPA) Ltd., London, 1991.
9. Cowderoy, A.J.C.; Jenkins, J.O.: New Trends in Cost-Estimation. in: Kitchenham; Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 63-88.
10. Fuchs, N.: COSMOS - Cost Management with Metrics of Specification. in: Fenton/Littlewood: Software Reliability and Metrics, Elsevier Applied Science, London New York, 1991, pp. 176-184.
11. Geringer, P.T.; Gullledge, T.R.; Hutzler, W.P.: Software Engineering Economics and Declining Budgets. Springer Publisher, Berlin Heidelberg New York, 1994.
12. Goodman, P.A.: Application of cost-estimation techniques: industrial perspective. Information and Software Technology, 34(1992)6, pp. 379-382.
13. Harrison, W.; Adrangi, B.: The Role of Programming Language in Estimating Software Development Costs. Journal of Management Information Systems, 3(1986-87)3, pp. 101-110.
14. Jeffery, D.R.: Time-Sensitive Cost Models in the Commercial MIS Environment. IEEE Transactions on Software Engineering, 13(1987)7, pp. 852-859.
15. Jones, C.: Applied Software Measurement. McGraw-Hill, Inc. 1991.
16. Jones, C.: Source of Errors in Software Cost Estimation. Technical Report, SPR Inc., Burlington, MA, Nov. 1993.
17. Kitchenham, B.A.: Empirical studies of assumptions that underlie software cost-estimation models. Information and Software Technology, 34(1992)4, pp. 211-218.
18. Kok, P.: New Approach to Software Cost Estimation. in: Fenton/Littlewood: Software Reliability and Metrics, Elsevier Applied Science, London New York, 1991, pp. 162-175.
19. Kok, P.A.M.; Kitchenham, B.A.; Kirakowski, J.: The MERMAID Approach to software cost estimation. in: ESPRIT'90, Kluner Academic Press, 1990, pp. 296-314.
20. Lederer, A.L.; Prasad, J.: Nine Management Guidelines for Better Cost Estimating. Comm. of the ACM, 35(1992)2, pp. 51-59.
21. MacDonell, S.G.: Comparative review of functional complexity assessment methods for effort estimation. Software Engineering Journal, May 1994, pp. 107-116.
22. Metelski, G.; Blair, T.: Application of Life Cycle Cost Modelling to New Generation of Telecommunication Products. Proceedings of the Quality Engineering Workshop, Bell Canada, October 15-17, 1991, pp. 19.1-19.10.
23. Miyazaki, Y.; Tarakado, M.; Ozaki, K.; Nozaki, M.: Robust Regression for Developing Software Estimation Models. The Journal of Systems and Software, 27(1994)1, pp. 3-1.
24. Parr, F.N.: An Alternative to the Rayleigh Curve Model for Software Development Effort. IEEE Transactions on Software Engineering, 6(1980)3, pp. 291-296.

25. Putnam, L.H.: A General Empirical Solution to the Macro Software Sizing and Estimating Problem. IEEE Transactions on Software Engineering, 4(1978)4, pp. 345-361.
26. Reiter, K.: Effort estimation methods for software development (german). in: Wirtschaftlichkeit von Software-Entwicklung und -Einsatz, Teubner Publisher, Stuttgart, German Chapter of ACM Berichte 36, 1992, pp. 197-212.
27. Royce, W.: Why software costs so much. IEEE Computer, May 1993, pp. 90-91 Sneed, H.M.: Estimating software development effort with the DATA POINTS method (german). Computer-Magazin, 11-12 (1991), pp. 41-46.
28. Srinivasan, B.; Martin, G.: MONSET - A Prototype Software Development Estimating Tool. Proc. of the Third Symposium on Assessment of Quality Software Development Tools, Washington D.C., June 7-9, 1994, IEEE Society Press, pp. 70-82.
29. Stricker, C.: Evaluating Effort Prediction Systems. Proceedings of the Tenth Annual Conference on Application of Software Metrics and Quality Assurance in Industry, Amsterdam, 29 September - 1 October 1993, Section 12.
30. Taff, L.M.; Borchering, J.W.; Hudgins, W.R.: Estimeetings: Development Estimates and a Front-End Process For a Large Project. IEEE Transactions on Software Engineering, 17(1991)8, pp. 839-849.
31. Tsuboi, N.: New Cost Model and Process Management Using that Model. Proceedings of the Third International Conference on Software Quality, Lake Tahoe, Nevada, 4-6 October 1993, pp. 215-222.
32. Verner, J.M.; Tate, G.: A Model for Software Sizing. The Journal of Software and Systems, 7 (1987), pp. 173-177.
33. Verner, J.M.; Tate, G.: A Software Size Model. IEEE Transactions on Software Engineering, 18(1992)4, pp. 265-277.
34. Vollmann, S.: Effort estimation in the software engineering (german). IWT Publisher, Munich, 1990.
35. Wakayama, H.; Takahashi, R.: A Software Development Cost Estimation Method Based on Software Classification. Proceedings of the Third International Conference on Software Quality, Lake Tahoe, Nevada, 4-6 October 1993, pp. 207-214.
36. Walston, C.E.; Felix, C.P.: A method of programming measurement and estimation. IBM System Journal, 1 (1977), pp. 54-75.
37. Warburton, R.D.H.: Managing and Predicting the Costs of Real-Time Software. IEEE Transactions on Software Engineering, 9(1983)5, pp. 562-569.
38. Wellman, F.: Software Costing. Prentice Hall Inc., 1992.
39. Zuse, H.: Prediction Models and Software Complexity Measures From a Measurement Theoretic View. Proceedings of the Third International Conference on Software Quality, Lake Tahoe, Nevada, 4-6 October 1993, pp. 49-58.

DATA (FLOW) MEASURES

1. Dhamdhere, D.M.; Khedker, U.P.: Complexity of Bidirectional Data Flow Analysis. ACM Symposium on Principles of Programming Languages, Charleston, January 10-13, 1993, pp. 397-408.
2. Munson, J.C.; Khoshgoftaar, T.M.: Measurement of Data Structure Complexity. The Journal of Systems and Software, 20(1993)3, pp. 217-225.
3. Stetter, F.: Software Engineering -- An Introduction (german). BI Wissenschaftsverlag, Mannheim, Wien, Zuerich, 4. Auflage, 1987.
4. Tai, K.: A Program Complexity Metric Based on Data Flow Information Control Graphs. Proceedings of the 7th International Conference on Software Engineering, October 1984, pp. 239-248.
5. Ural, H.; Yang, B.: Modeling Software for Accurate Data Flow Representation. Proceedings of the 15th International Conference on Software Engineering, May 17-21, Baltimore, 1993, pp. 277-286.

EVALUATION

1. Burgess, A.: Mad about or mad at measurement? IEEE Software, January 1995, pp. 115-116.
2. Deininger, M.: A method for the classification of software metrics (Ein Schema zur Klassifikation von Metriken). Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, Oktober 1990, pp. 15-29.
3. Denvir, T.; Herman, R.; Whitty, R.W.: Formal Aspects of Measurement. Springer Verlag, 1992.
4. Dumke, R.: Analysis and Evaluation of Software Measures (german). Wiss. Zeitschrift der TU Magdeburg, 32(1988)3, pp. 77-81.
5. Ebert, C.: Visualization Techniques for Analyzing and Evaluation Software Measures. IEEE Transactions on Software Engineering, 18(1992)11, pp. 1029-1034.
6. Ejiogu, L.O.: An Introduction to the Introduction of Software Metrics. SOFTMETRICS Inc., Working Paper, Chicago, January 1993.
7. Gustafson, D.A.; Prasad, B.: Properties of Software Measures. in: Denvir et. al.: Formal Aspects of Measurement. Springer Verlag, 1992, pp. 179-193.
8. Itzfeldt, W.D.: Methodological Requirements to software measures (german). Angewandte Informatik, 25(1983)2, pp. 55-61.
9. Jones, C.: Software metrics: Good, bad, and missing. IEEE Computer, September 1994, pp. 98-100.
10. Lakshmanan, K.B.; Jayaprakash, S.; Sinha, P.K.: Properties of Control-Flow Complexity Measures. IEEE Transactions on Software Engineering, 17(1991)12, pp. 1289-1295.
11. Melton, A.: Specifying Internal, External, and Predictive Software Metrics. in: Denvir et. al.: Formal Aspects of Measurement. Springer Verlag, 1992, pp. 194-208.

12. Ostrand, T.J.; Weyuker, E.J.: Collecting and Categorizing Software Error Data in an Industrial Environment. *The Journal of Systems and Software*, 4 (1984), pp. 289-300.
13. Perlis, A.; Sayward, F.; Shaw, M.: *Software Metrics: An Analysis and Evaluation*. The MIT Press, 1983.
14. Prather, R.E.: An Axiomatic Theory of Software Complexity Measure. *The Computer Journal*, 27(1984)4, pp. 340-347.
15. Redish, K.A.; Smyth, W.F.: Evaluating Measures of Program Quality. *The Computer Journal*, 30(1987)3, pp. 228-232.
16. Shen, V.Y.: Using metrics in quality management. *IEEE Software*, July 1990, pp. 80-85
Shepperd, M.: An evaluation of software product metrics. *Information and Software Technology*, 30(1988)3, pp. 177-188.
17. Tian, J.; Porter, A.; Zelkowitz, M.V.: An Improved Classification Tree Analysis of High Cost Moduls Based an Axiomatic Definition of Complexity. *Proceedings of the Third International Symposium on Software Reliability Engineering*, Research Triangle Park, NC, October 8-9, pp. 164-172.
18. Tian, J.; Zelkowitz, M.V.: A Formal Program Complexity Model and its Application. *The Journal of Systems and Software*. 17(1992), pp. 253-266.
19. Weyuker, E.J.: Evaluating Software Complexity Measures. *IEEE Transactions on Software Engineering*, 14(1988)9, pp. 1357-1365.
20. Zelkowitz, M.V.; Tian, J.: A Theory of Prime Program Complexity. University of Maryland, October 1992.
21. Zuse, H.: Discussion of the Weyuker Properties from a Measurement Theoretic View. Submitted to the *IEEE Transactions on Software Engineering*
22. Zuse, H.: Properties of Software Metrics. *Proceedings of the International Software Quality Conference*, Dayton, Ohio, October 6-9, 1991, pp. 12-18.
23. Zuse, H.: Software Complexity Measures. *Proceedings of the Workshop Computer Aided Software Evaluation*, TU Magdeburg, October 1990, pp. 110-128.
24. Zuse, H.; Bollmann, P.: Using Measurement Theory to Describe the Properties and Scales of Static Software Complexity Metrics. *SIGPLAN Notices*, 24(1989)8, pp. 23-33.

EVALUATION/CERTIFICATION MEASURES

1. Anderson, E.E.: A Heuristic for Software Evaluation and Selection. *Software -- Practice and Experience*, 19(1989)8, pp. 707-717.
2. Card, D.N.; McGarry, F.E.; Page, G.T.: Evaluating Software Engineering Technologies. *IEEE Transactions on Software Engineering*, 13(1987)7, pp. 845-851.
3. Coallier, F.: A Method for the Assessment of Telecom Software System Development Capability. *Quality Engineering Workshop*, Ottawa, October 16-17, 1991.

4. Dumke, R.: Computer aided software evaluation (german). Research Report, TU Magdeburg, March 1990.
5. Grogono, P.; Preece, A.; Shinghal, R., Suen, C.Y.: Techniques for Evaluating Expert Systems in Telecommunications. Quality Engineering Workshop, Ottawa, October 16-17, 1991.
6. Mosley, V.: How to Assess Tools Efficiently and Quantitatively. IEEE Software, May 1992, pp. 29-32.
7. Mueller, K.: Evaluation of CASE Tools (german).Bewertung von Study, TU Magdeburg, July 1992.
8. Ramsey, C. L.; Basili, V.R.: An Evaluation of Expert Systems for Software Engineering Management. IEEE Transactions on Software Engineering, 15(1989)6, pp. 747-759.
9. Robillard, P.N.; Coupal, D.; Mayrand, J.: Software Evaluation Based on Static Metrics. Bell Canada Quality Engineering Workshop, Montreal, October 4-5, 1990.
10. Schneider, K.: Systematic evaluation of CASE tools (german). Proceedings of the Tool'91, Karlsruhe 1991, pp. 263-278.
11. Software -- Guetesicherung (Software quality assurance RAL-GZ 901. Guetegemeinschaft Software e.V., Frankfurt am Main, November 1985.
12. Suilmann, M.: Computer aided evaluation of CASE tools (german). Proceedings of the TOOL'91, Karlsruhe 1991, pp. 351-358.
13. Voas, J.; Payne, J.; Miller, K.: A Future Direction for Software Certification Testing Using Sensitivity Analysis. Proceedings of the International Software Quality Conference, Dayton, Ohio, 1991, pp. 202-207.
14. Welzel, D.; Hausen, H.: Metric-Based Software Evaluation Method. First European International Conference on Software Testing, Analysis & Review (EuroStar), London, October 25-28, 1993, pp. 429-438.
15. Wingrove, A.A.: Software Certification. in: Fenton, N.; Littlewood, B.: Software Reliability and Metrics. Elsevier Applied Science, London New York, 1991, pp. 192-216.
16. Wise, S.: Software evaluation - lessons from the GIS evaluation. University Computing, (1991) 13, pp. 16-20.
17. Dumke, R.; Winkler, A.: Requirements to and Comparing Analysis of Measurement Tools. (german) in: Dumke/Zuse: Theorie und Praxis der Softwaremessung, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp. 248-287.

FACTOR ANALYSIS

1. Coupal, D.; Robillard, P.N.: Factor Analysis of Source Code Metrics. The Journal of Systems and Software, 12 (1990), pp. 263-269.
2. Coupal, D.; Robillard, P.N.: How meaningful are software metrics? Bell Canada Quality Engineering Workshop, Montreal, October 4-5, 1990.

3. Mata-Toledo, R.A.; Gustafson, D.A.: A Factor Analysis of Software Complexity Measures. *The Journal of Systems and Software*, 17(1992), pp. 267-273.

FORMAL SPECIFICATION

1. Bainbridge, J.; Whitty, R.W.; Wordsworth, J.: Obtaining Structural Metrics of Z Specifications for System Development. 5th Annual Z User Meeting, 1990.
2. Cardenes, S.; Zelkowitz, M.V.: Evaluation Criteria for Functional Specification. *Proceedings of the 12th International Conference on Software Engineering*, March~26-30, Nice, France, 1990, pp. 26-33.
3. Coulter, N.S.; Cooper, R.B.; Solomon, M.K.: Information-theoretic Complexity of Program Specification. *The Computer Journal*, 30(1987)3, pp. 223-227.
4. Fuchs, N.; Stainer, S.: Language Independent Definition of Axiomatic Metrics. in: Denvir et. al.: *Formal Aspects of Measurement*. Springer Verlag, 1992, pp. 84-107.
5. Kuhn, D.R.: A Technique for Analyzing the Effects of Changes in Formal Specifications. *The Computer Journal*, 35(1992)6, pp. 574-578.
6. Melhart, B.E.: A Continuum of Formal Methods for Specification of Quality Software. *Proceedings of the International Software Quality Conference*, Dayton, Ohio, 1991, pp. 66-71.
7. Mitchell, J.; Urban, J.E.; McDonald, R.: The Effect of Abstract Data Types on Program Development. *IEEE Computer*, August 1987, pp. 85-88.
8. Moore, A.P.: The Specification and Verified Decomposition of System Requirements Using CSP. *IEEE Transactions on Software Engineering*, 16(1990)9, pp. 932-948.
9. Richter, S.: Investigations to the applicability of software metrics for specification languages of communication systems (german). Master's Thesis, TU Magdeburg, 1992.
10. Shaw, R.: The role of formal methods in managing complexity. in Mitchell, R.J.: *Managing Complexity in Software Engineering*. IEEE Computing Series 17, London, 1990, pp. 51-86.
11. VanSuetendael, N.; Elwell, D.: *Software Quality Metrics*. Technical Report, Atlantic City, New Jersey, Juli 1991.
12. Whitty, R.W.: Research in specification metrics. Paper 6, IEE Colloquium of Software Metrics, Digest No. 1990/003, 1990.
13. Whitty, R.W.: Structural Metrics for Z Specifications. in the *Proceedings of the Z User Workshop*, Oxford 1989, Springer Verlag, 1990, pp. 186-191.

FUNCTION POINTS

1. Albrecht, A.J.: *Measuring Application Development Productivity*. Tutorial -- Programming Productivity: Issues for the Eighties, IEEE Computer Society, ISBN 0-8186-0681-9, 1986, pp. 35-44.

2. Albrecht, A.J.; Gaffney, J.E.: Software Function, Source Lines of Code, and Development Effort Prediction: A Software Science Validation. IEEE Transactions on Software Engineering, 9(1983)6, pp. 639-648.
3. Abran, A.; Robillard, P.N.: Identification of the structural weaknesses of Function Point metrics. Proceedings of the 3rd Annual Oregon Workshop on Software Metrics, Portland, Oregon, March 18-19, 1991.
4. Abran, A.; Robillard, P.N.: Software Management Based on Software Deliverables. Proceedings of the CIPS/CATA Congress 90, Ottawa Ontario, Canada, May 16-18 1990, pp. 237-245.
5. Behrens, C.A.: Measuring the Productivity of Computer Systems Development Activities with Function Points. IEEE Transactions on Software Engineering, 9(1983)6, pp. 648-652.
6. Frach, K.: Complexity and effort in the development of a large scale software project (german). Study, IBM Hamburg, Technical University of Magdeburg, 1993.
7. Gaffney, J.E.: The Impact on Software Development Costs of Using HOL's. IEEE Transactions on Software Engineering, 12(1986)3, pp.~496-499.
8. Grossjohann, R.: Significance of the Function Point Method under Recession. in: Dumke/Zuse: Theorie und Praxis der Softwaremessung, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp.20-34.
9. Huerten, R.: Man month and lines of code are secondary measures (german). Computerwoche, 46(1992) Nov., pp. 13-14.
10. Hufschmidt, B.: What is the International Function Point Users Group (IFPUG)? METRICVIEWS, Newsletter of the International Function Point Users Group, Westerville, Ohio, July 1992.
11. Jones, C.: Table of Programming Languages and Levels. Technical Report, SPR Inc., Burlington, MA, January 1994.
12. Kemerer, C.F.; Porter, B.S.: Improving the Reliability of Function Point Measurement: An Empirical Study. IEEE Transactions on Software Engineering, 18(1992)11, pp. 1011-1024.
13. Kitchenham, B.: Using Function Points for Software Cost Estimation - Some Empirical Results. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29 September - 1 October 1993.
14. Kitchenham, B.; Kaensaelae, K.: Iter-item Correlations among Function Points. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 11-14.
15. Low, G.C.; Jeffery, D.R.: Function Points in the Estimation and Evaluation of the Software Process. IEEE Transactions on Software Engineering, 16(1990)1, pp. 64-71.
16. Miluk, G.: Introduction to Function Points. Proceedings of the International Software Quality Conference, Dayton, Ohio, 1991, pp. 89-94.

17. Nishiyama, S.; Furuyama, T.: The validity and applicability of function point analysis - as related to specification quality and ergonomics. Proc. of the Fourth European Conference of Software Quality, October 17-20, Basel, Switzerland, pp. 479-490.
18. Onvlee, J.: Use of Function Points for Estimation and Contracts. Proceedings of the Tenth Annual Conference on Application of Software Metrics and Quality Assurance in Industry, Amsterdam, 29 September - 1 October 1993, Section 13.
19. Shepperd, M.: Some Observations on Function Points. Proc. of the 11th CSSR Conference on Software Evolution, Models and Metrics, September 7-9, Dublin, Ireland, Section 21.
20. Symons, C.R.: Function Point Analyses: Difficulties and Improvements. IEEE Transactions on Software Engineering, 14(1988)1, pp. 2-11.
21. The function point method and his application (german). Volkswagen AG, Wolfsburg 1989.

FUNCTIONAL PROGRAMMING

1. Berg, Klaas van den : Software Measurement and Functional Programming. Ph.D., University of Twente, Netherlands, 1995.
2. Howden, W.E.: Functional Program Testing. IEEE Transactions on Software Engineering, 6(1980)2, pp. 162-169.
3. Joy, M.; Axford, T.: A Standard for a Graph Representation for Functional Programs. SIGPLAN Notices, 23(1988)1, pp. 75-82.
4. Ryder, B.G.: Constructing the Call Graph of a Program. IEEE Transactions on Software Engineering, 5(1979)3, S,~216-226.
5. Stoeffler, K.: Investigations to software metrics for nonprocedural languages (german). Master's Thesis, TU Magdeburg, 1992.

HALSTEADS SOFTWARE SCIENCE

1. Baker, A.L.; Zweben, S.H.: The Use of Software Science in Evaluating Modularity Concepts. IEEE Transactions on Software Engineering, 5(1979)2, pp. 110-120.
2. Bandyopadhyay, S.K.: A Study on program level dependency of implemented algorithms on its potential operands. SIGPLAN Notices, 16(1981)2, pp. 18-25.
3. Bandyopadhyay, S.K.: Theoretical Relationships Between Potential Operands and Basic Measurable Properties of Algorithm Structure, SIGPLAN Notices, 16(1981)2, pp. 26-34.
4. Card, D.N.; Agresti, W.W.: Resolving the Software Science Anomaly. The Journal of Systems and Software, (1987)7, pp. 29-35.
5. Christensen, K.; Fitsos, G.P.; Smith, C.P.: A perspective on software science. IBM System Journal, 20(1981)4, pp. 372-387.
6. Comer, D.; Halstead, M.H.: A Simple Experiment in Top-Down Design. IEEE Transactions on Software Engineering, 5(1979)2, pp. 105-109.

7. Coulter, N.S.: Software Science and Cognitive Psychology. IEEE Transactions on Software Engineering, 9(1983)2,4 pp. 166-171.
8. Curtis, B.; Sheppard, S.B.; Millman P.: Third Time Charm: Stronger Prediction of Programmer Performance by Software Complexity Metrics. Proceedings of the 4th International Conference on Software Engineering, September 17-19, Munich, 1979, pp. 356-360.
9. Curtis, B.; Sheppard, S.B.; Millman, P.; Borst, M.A.; Love, T.: Measuring the Psychological Complexity of Software Maintenance Tasks with the Halstead and McCabe Metrics. IEEE Transactions on Software Engineering, 5(1979)2, pp. 96-104.
10. Gordon, R.D.: A Qualitative Justification for a Measure of Program Clarity. IEEE Transactions on Software Engineering, 6(1979)2, pp. 121-128.
11. Halstead, M.H.: Elements of Software Science. Prentice-Hall, Inc., New York, 1977.
12. Hamer, P. G.; Frewin, G.D.: M.H. Halstead's Software Science -- A Critical Examination. Proceedings of the 6th International Conference on Software Engineering, Sept. 13-16, 1982, Tokyo, Japan, pp. 197-206.
13. Hegewald, H.; Seibt, A.: Investigations to the Halstead Metric (Untersuchungen zur Halstead-Metrik). Study, TU Magdeburg, 1987.
14. Jensen, H.A.; Vairavan, K.: An Experimental Study of Software Metrics for Real-Time Software. IEEE Transactions on Software Engineering, 11(1985)2, pp. 231-234.
15. Keller-McNulty, S.; McNulty, M.S.; Gustafson, D.A.: Stochastic Models for Software Science. The Journal of Systems and Software, 12 (1990), pp. 59-68.
16. Khoshgoftaar, T.M.; Munson, J.C.; Lanning, D.L.: Alternative Approaches for the Use of Metrics to Order Programs by Complexity. The Journal of Systems and Software, 24(1994)3, pp. 211-221.
17. Konstam, A.H.; Wood, D.E.: Software Science Applied to APL. IEEE Transactions on Software Engineering, 11(1985)10, pp. 994-1000.
18. Lassez, J.L.; Shepherd, K.J.; Lassez, C.: A Critical Examination of Software Science. The Journal of Systems and Software, (1981)2, pp. 105-112.
19. Mehndiratta, B.; Grover, P.S.: Software Metrics -- An Experimental Analysis. SIGPLAN Notices, 25(1990)2, pp. 35-41.
20. Ottenstein, L.M.: Quantitative Estimates of Debugging Requirements. IEEE Transactions on Software Engineering, 5(1979)5, pp. 504-514.
21. Ramamurthy, B.; Melton, A.: A Synthesis of Software Science Measures and the Cyclomatic Number. IEEE Transactions on Software Engineering, 14(1988)8, pp. 1116-1121.
22. Shaw, W.H.; Howatt, J.W.; Maness, R.S.; Miller, D.M.: A Software Science Model of Compile Time. IEEE Transactions on Software Engineering, 15(1989)5, pp. 543-549.

23. Shen, V.Y.; Conte, S.D.; Dunsmore, H.E.: Software Science Revisited: A Critical Analysis of the Theory and its Empirical Support. IEEE Transactions on Software Engineering, 9(1983)2, pp. 155-165.
24. Stetter, F.: A Measure of Program Complexity. Computer Languages, 9(1984)3/4, pp. 203-208.
25. Stetter, F.: A simplyfied model of software science (german.) Angewandte Informatik, 26(1984)4, pp. 147-151.
26. Stetter, F.: Estimates in software science. Informatik Bericht Nr.31, Fernuniversitaet Hagen, 1983.
27. Woodfield, S.N.: An Experiment on Unit Increase in Problem Complexity. IEEE Transactions on Software Engineering, 5(1979)2, pp. 76-79.
28. Zweben, S.H.; Halstead, M.H.: The Frequency Distribution of Operators in PL/I Programs. IEEE Transactions on Software Engineering, 5(1979)2, pp. 91-95.

HYBRID MEASURES

1. Basili, V.R.; Hutchens, D.V.: An Empirical Study of a Syntactic Complexity Family. IEEE Transactions on Software Engineering, 9(1983)6, pp. 664-672.
2. Elshoff, J.L.: Characteristic Program Complexity Measures. IEEE Proceedings of the 7th International Conference on Software Engineering, March 26-29, 1984 Orlando, Florida, pp. 288-293.
3. Harrison, W.; Cook, C.R.: A Note on the Berry-Meekings Style Metric. Comm. of the ACM, 29(1986)2, pp. 123-125.

HYPERGRAPH MEASURES

1. Stetter, F.: Modelling programs by the notion of hypergraph. Software Engineering Journal, March 1989, pp. 107-111.
2. Stetter, F.; Rooyen, H.O.van: Program Measures Based on a Graph-Like Model. EIK 29, Germany, (1993)1, pp. 55-75.

HYPERTEXT METRICS

1. Rivlin, E.; Botafogo, R.; Schneiderman, B.: Navigating in Hyperspace: Designing a Structure-Based Toolbox. Comm. of the ACM, 37(1994)2, pp. 87-96.

INFORMATION MEASURES

1. Baumann, P.: Software Evaluation -- A Semantical Approach to Information Measures (german). Springer Verlag, 1991.
2. Baumann, P.; Richter, L.: Which expressiveness have the actual software metrics? (german). Wirtschaftsinformatik, 34(1992)6, pp. 624-631.
3. Harrison, W.: An Entropy-Based Measure of Software Complexity. IEEE Transactions on Software Engineering, 18(1992)11, pp. 1025-1029.

4. Henry, S.; Kafura, D.: Software Structure Metrics Based on Information Flow. IEEE Transactions on Software Engineering, 7(1981)5, pp. 510-518.
5. Henry, S.; Kafura, D.: The Evaluation of Software Systems' Structure Using Quantitative Software Metrics. Software -- Practice and Experience, 14(1984)6, pp. 561-573.
6. Henry, S.; Selig, C.: Predicting Source-Code Complexity at the Design Stage. IEEE Software, March 1990, pp. 36-44.
7. Kafura, D.; Henry, S.: Software Quality Metrics Based on Interconnectivity. The Journal of Systems and Software, 2 (1982), pp. 121-131.
8. Samadzadeh-Hadidi, M.: Measurable Characteristics of the Software Development Process Based on a Model of Software Comprehension. Ph. Dissertation, University of Southwestern Louisiana, 1987.

LOGICAL PROGRAMMING

1. Belli, F.; Jack, O.: PROTest II: Testing Logic Programs. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 128-135.
2. Luo, G.; Bochmann, G.v.; Sarikaya, B.; Boyer, M.: Control-Flow Based Testing of Prolog Programs. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, 1992, pp. 104-113.
3. Markusz, Z.; Kaposi, A.A.: Complexity Control in Logic-based Programming. The Computer Journal, 28(1985)5, pp. 487-495 see also Stoeffler.

MAINTENANCE METRICS

1. Abran, A.; Nguyenkim, H.: Analysis of Maintenance Work Categories Through Measurement. Proceedings of the Conference on Software Maintenance 1991, Sorrento, Italy, October 15-17, pp. 104-113.
2. Coleman, D.; Ash, D.; Loather, B.; Oman, P.: Using Metrics to Evaluate Software System Maintainability. IEEE Computer, August 1994, pp. 44-49.
3. Coleman, D.; Lowther, B.; Oman, P.: The Application of Software Maintainability Models in Industrial Software Systems. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1993, Silver Falls, Oregon.
4. Debou, C.; Stainer, S.: Improving the maintenance process: a quantitative approach. Proceedings of the 6th International Conference on Software Engineering and its Applications, Paris, November 1993, pp. 273-282.
5. Dumke, R.; Ziehm, W.: Software Measurement in Data Base Systems. Internationale Conference on Data base, TH Magdeburg, January 1986.
6. Foster, J.: Program Lifetime: A Vital Statistic for Maintenance. Proceedings of the Conference on Software Maintenance 1991, Sorrento, Italien, October 15-17, pp. 98-103.

7. Frewin, G.D.: Metrics in Procurement -- a discussion paper. in: Kitchenham;Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp.~89-102.
8. Genuchten M. van; Brethouwer, G.; Booman, T. van; Heemstra, F.: Empirical study of software maintenance. Information and Software Technology, 34(1992)8, pp. 507-512.
9. Georgiadon, E.G.; Sadler, C.J.: Testing Restructured Software. First European International Conference on Software Testing, Analysis & Review (EuroStar), London, October 25-28, 1993, pp. 139-148.
10. Harrison, W.; Cook, C.: Insights on Improving the Maintenance Process Through Software Measurement. Proceedings of the International Conference on Software Maintenance, San Diego, November 26-29, 1990, pp. 37-44.
11. Kafura, D.; Reddy, G.R.: The Use of Software Complexity Metrics in Software Maintenance. IEEE Transactions on Software Engineering, 13(1987)3, pp. 335-343.
12. Lehner, F.: Software Maintenance (german). Carl Hanser Verlag Munich Vienna, 1991
Lehner, F.; Hofmann, H.F.; Setzer, R.: Maintenance of knowledge data basis (german). WHU (The Koblenz School of Corporate Management) Research Paper, Nr. 15, Koblenz, Germany, 1992.
13. Littlewood, B.; Miller, D.R.: A Conceptual Model of the Effect of Diverse Methodologies on Coincident Failures in Multi-version Software. in: Kitchenham;Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 321-334.
14. Lucas, J.: A tool for the visualisation of modules for the maintenance (german). Proceedings of the TOOL'91, Karlsruhe 1991, pp. 411-422.
15. Malaiya, Y.K.; Mayrhauser, A. von; Srimani, P.K.: The Nature of Fault Exposure Ration. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, 1992, pp. 23-32.
16. Oman, P.; Hagemeister, J.: Construction and Testing of Polynimials Predicting Software Maintainability. The Journal of Systems and Software, 24(1994)3, pp. 251-266.
17. Oman, P.; Hagemeister, J.: Metrics for Assessing a Software System`s Maintainability. Proceedings of the Conference on Software Maintenance, Orlando, Nov. 9-12 1992, pp. 337-344.
18. Penzel, H.: A Strategy Leading to a Cost Oriented Information Management of Maintenance. in: Thurner, R.: Reengineering -- Ein integrales Wartungskonzept zum Schutz von Software-Investitionen, AIT Publisher, 1991, pp. 223-248.
19. Pickard, M.M.; Carter, B.D.: Maintainability: What Is It And How Do We Measure It? Software Engineering Notes, 18(1993)3, pp.~A-36 - A39.
20. Poppe, S.: Application of Software Metrics in Software Maintenance (german). dissertation degree, TU Magdeburg , 1993.
21. Porter, A.A.: Using Measurement-Driven Modeling to Provide Empirical Feedback to Software Developers. The Journal of Systems and Software, 20(1993)3, pp. 237-243.

22. Rombach, H.D.: A Controlled Experiment on the Impact of Software Structure on Maintainability. *IEEE Transactions on Software Engineering*, 13(1987)3, pp.~344-354.
23. Schaefer, H.: Software Quality Assurance in the Maintenance Phase. Second European Conference on Software Quality Assurance, Conf.Proc., Oslo, 1990.
24. Schneidewind, N.F.: The State of Software Maintenance. *IEEE Transactions on Software Engineering*, 13(1987)3, pp. 303-310.
25. Shepperd, M.; Ince, D.: Controlling Software Maintainability. Second European Conference on Software Quality Assurance, Conf.Proc., Oslo, 1990.
26. Silverman, B.G.: Software Cost and Productivity Improvements: An Analogical View. *IEEE Computer*, May 1985, pp. 86-95.
27. Stark, G.E.; Kern, L.C.; Vowell, C.W.: A Software Metric Set for Program Maintenance Management. *The Journal of Systems and Software*, 24(1994)3, pp. 239-249.
28. Yau, S.S.; Nicholl, R.A.; Tsai, J.J.; Liu, S.S.: An Integrated Life-Cycle Model for Software Maintenance. *IEEE Transactions on Software Engineering*, 14(1988)8, pp. 1128-1144.
29. Zhuo, F.; Lowther, B.; Oman, P.; Hagemester, J.: Constructing and Testing Software Maintainability Assessment Models. *Proceedings of the First International Software Metrics Symposium*, Baltimore, May 21-22, 1993, pp. 61-70.
30. Zuse, H.: Criteria for Program Comprehension Derived from Software Complexity Metrics. *Proceedings of the 2nd Workshop on Program Comprehension*, July 8-9, Capri, 1993, pp. 8-16.
31. Zuse, H.: Measuring Factors Contributing to Software Maintenance Complexity. *Proceedings of the Second International Conference on Software Quality*, Research Triangle Park, NC, October 5-7, pp. 178-190.

MANAGEMENT

1. Azuma, M.; Mole, D.: Software Management Practice and Metrics in the European Community and Japan: Some Results of a Survey. *The Journal of Systems and Software*, 26(1994)1, pp. 5-18.
2. Boehm, B.W.: Industrial software metrics top 10 list. *IEEE Software*, (1987) September, pp. 84-85.
3. Bourguignon, J.P.: Structuring for managing complexity. in Mitchell, R.J.: *Managing Complexity in Software Engineering*. IEE Computing Series 17, London, 1990, pp. 217-224.
4. Bradley, L.: Monitoring Developer-Centered SQA for the Analysis and Design Stages. *Proceedings of the International Software Quality Conference*, Dayton, Ohio, October 7-9, 1991, pp. 128-131.
5. Fenick, S.: Implementing Management Metrics: An Army Program. *IEEE Software*, March 1990, pp. 65-73.

6. Geringer, P.T.; Gulledge, T.R.; Mutzler, W.P.: Software Engineering Economics and Declining Budgets. Springer Publ., Berlin Heidelberg New York, 1994.
7. Kitchenham, B.A.: Measuring to manage. in Mitchell, R.J.: Managing Complexity in Software Engineering. IEE Computing Series 17, London, 1990, pp. 153-166.
8. Knoell, H.D.: How to Involve the Enduser Into Quality Assurance. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 16-22.
9. Maiocchi, M.; Marchetti, B.; Pina, D.: Software Quality, Risk and Costs: A Proposed Framework. Proceedings of the Second European Conference on Software Quality Assurance, Oslo, 1990.
10. Miller, S.E.; Tucker, G.T.: Software Development Process Benchmarking. Proc. of the IEEE Global Telecommunications Conference, December 2-5, 1991, pp. 153-157.
11. Norris, M.; Rigby, P.; Payne, M.: The Healthy Software Project - A Guideline to Successful Development and Management. John Wiley & Sons, 1993.
12. Porter, A.A.; Selby, R.W.: Evaluating Techniques for Generating Metric-based Classification Trees. The Journal of Systems and Software, 12 (1990), pp. 209-218.
13. Sroka, J.V.; Gosling, C.A.: Using Quantitative Activity Models in Project Management. in: Kitchenham; Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 181-200.
14. Stark, G.; Durst, R.C.; Vowell, C.W.: Using Metrics in Management Decision Making. IEEE Computer, September 1994, pp. 42-48.
15. Tolochko, S.L.: The Readiness Growth Model: A Quantitative Analysis of Software Risk. Proceedings of the First European Conference on Software Testing, Analysis & Review (EuroStar), London, October 25-28, 1993, pp. 409-427.
16. Weller, E. F.: Using Metrics to Manage Software Projects. IEEE Computer, September 1994, pp. 27-33.

MANAGEMENT BIBLIOGRAPHY

1. Abbott "Program Design by Informal English Descriptions," Communications of the ACM, Vol. 26, No. 11, November 1983, pp. 882 - 894.
2. Abdel-Hamid and S. Madnick, "Lessons Learned from Modeling the Dynamics of Software Project Management," Communications of the ACM, December 1989.
3. Abdel-Hamid and S. Madnick, Software Project Dynamics, An Integrated Approach, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
4. Association for Computing Machinery, Special Issue of SIGPLAN Notices on the Object-Oriented Programming Workshop, Vol. 21, No. 10, October 1986.
5. Association for Computing Machinery, OOPSLA '86 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986.

6. Association for Computing Machinery, OOPSLA '87 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987.
7. Association for Computing Machinery, OOPSLA '87 Addendum to the Proceedings, Special Issue of SIGPLAN Notices, Vol. 23, No. 5, May 1988.
8. Association for Computing Machinery, OOPSLA '88 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988.
9. Association for Computing Machinery, OOPSLA '89 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989.
10. Association for Computing Machinery, OOPSLA/ECOOP '90 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990.
11. Association for Computing Machinery, OOPSLA/ECOOP '91 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 26, No. 11, November 1991.
12. Ahmed, A. Wong, D. Sriam, and R. Logcher, A Comparison of Object-Oriented Database Management Systems for Engineering Applications, Research Report No. R91-12, Order Number IESL90-03, 91-03, Massachusetts Institute of Technology, Department of Civil Engineering, Cambridge, Massachusetts, May 1991.
13. Anderson, J. McDonald, L. Holland, and E. Scranage, "Automated Object-Oriented Requirements Analysis and Design," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 265 - 272.
14. American National Standards Institute/Institute of Electrical and Electronics Engineers, IEEE Standard for Software Verification and Validation Plans (ANSI/IEEE Standard 1012-1986), Institute of Electrical and Electronics Engineers, New York, New York, 1987.
15. American National Standards Institute/Institute of Electrical and Electronics Engineers, IEEE Standard of Software Project Management Plans (ANSI/IEEE Standard 1058.1-1987), Institute of Electrical and Electronics Engineers, New York, New York, 1988.
16. American National Standards Institute/Institute of Electrical and Electronics Engineers, IEEE Standard for Software Quality Assurance Plans (IEEE Standard 730-1989), Institute of Electrical and Electronics Engineers, New York, New York, 1990.
17. American National Standards Institute/Institute of Electrical and Electronics Engineers, IEEE Standard for Software Configuration Management Plans (IEEE Standard 828-1990), Institute of Electrical and Electronics Engineers, New York, New York, 1991.
18. Anthes, "Slashing Away At Sacred Cows," ComputerWorld, Vol. 26, No. 14, April 6, 1992, pp. 115 - 117.
19. Arnold, et al., An Evaluation of Five Object-Oriented Development Methods - Report HPL-91-52, HP Laboratories, Bristol, United Kingdom, 1991.
20. Atkinson, F. Bancilhon, D. DeWitt, K. Dittrich, D. Maier, and S. Zdonik, "The Object-Oriented Database System Manifesto," (Invited Paper), Proceedings of the First International Conference on Deductive and Object-Oriented Databases, Kyoto, Japan, December 4-6, 1989, pp. 40 - 57.

21. Badiru and G. Whitehouse, Computer Tools, Models, and Techniques for Project Management, Tab Books, Blue Ridge Summit, Pennsylvania, 1989.
22. Baker, D. Murphy, and D. Fisher, "Factors Affecting Project Management Success," in Project Management Handbook, D. Cleland and W. King, Editors, Van Nostrand Reinhold, 1983, New York, New York, pp. 669 - 685.
23. Baskette, "Life Cycle Analysis of an Ada Project," IEEE Software, Vol. 4, No. 1, 1987, pp. 40 - 47.
24. Gordon Bell, High-Tech Ventures, Addison Wesley, Reading, Massachusetts, 1991.
25. Bennett, "Accepting object technology," The Hotline on Object-Oriented Technology, SIGS Publications, New York, New York, Vol. 3, No. 2, pp. 1 - 4.
26. Berard, "Object-Oriented Requirements Analysis," Hotline On Object-Oriented Technology, Vol. 1, No. 8, June 1990, pp. 9 - 11.
27. Berard, Essays on Object-Oriented Software Engineering, Volume 1, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
28. Berard, Selecting and Using Consultants for Object-Oriented Technology," Journal of Object-Oriented Programming, Vol. 6, No. 5, September 1993, pp. 48 - 53.
29. Bertino and L. Martino, "Object-Oriented Database Management Systems: Concepts and Issues," IEEE Computer, Vol. 24, No. 4, April 1991, pp. 33 - 47.
30. Bézivin and B. Meyer, Editors, Technology of Object-Oriented Languages and Systems: Tools 4, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
31. Bézivin, J.-M. Hullot, P. Cointe, and H. Lieberman, ECOOP '87: Proceedings of the European Conference on Object-Oriented Programming, Lecture Notes on Computer Science, Volume 276, Springer Verlag, New York, New York, 1987.
32. Birchenough and J.R. Cameron, "JSD and Object-Oriented Design," JSP & JSD: The Jackson Approach to Software Development, IEEE Computer Society Press, Washington, D.C., 1989.
33. Blair, J. Gallagher, D. Hutchison, and D. Sheperd, Object-Oriented Languages, Systems and Applications, Halsted Press, New York, New York, 1991.
34. Bochenski, "On Object-Oriented Programming, Databases," Software, Vol. 8, No. 11, September 1988, page 42.
35. Boehm, Software Engineering Economics, Prentice Hall, Englewood Cliffs, New Jersey, 1981.
36. Boehm, "A Spiral Model of Development and Enhancement," Software Engineering Notes, Vol. 11, No. 4, August, 1986, pp. 14 - 24.
37. Booch, "Object Oriented Design," Ada Letters, Vol. I, No. 3, March- April 1982, pp. 64 - 76.

38. Booch, Software Engineering with Ada, Benjamin Cummings, Menlo Park, California, 1983.
39. Booch, "Object Oriented Development," IEEE Transactions on Software Engineering, Vol. SE-12, No. 2, February 1986, pp. 211 - 221.
40. Booch, "On the Concepts of Object-Oriented Design," in Modern Software Engineering: Foundations and Current Perspectives, P.A. Ng and R.T. Yeh, Editors, Van Nostrand Reinhold, New York, New York, 1990, pp. 165 - 204.
41. Booch, Object-Oriented Design With Applications, Benjamin/Cummings, Redwood City, California, 1991.
42. Booch, "Object-Oriented Design," OOPSLA '91 Tutorial Notes, October 6-11, 1991.
43. Booch, Object-Oriented Analysis and Design With Applications, Second Edition, Benjamin/Cummings, Menlo Park, California, 1994.
44. Booch, "Coming of Age In an Object-Oriented World," IEEE Software, Vol. 11, No. 6, November 1994, pp. 33 - 41.
45. Boone and V. Merlyn, "Getting the Process Right," ComputerWorld, Vol. 25, No. 26, June 10, 1991, pp. 75 - 77.
46. Bouldin, Agents of Change. Yourdon Press/Prentice Hall, Englewood Cliffs, New Jersey, 1989.
47. Bozman, "Firms Move Into the OOP Lane," ComputerWorld, Vol. XXIV, No. 30, July 23, 1990, pp. 23, 31.
48. Bozman, "They're Still Saying OOPs," ComputerWorld, Vol. XXIV, No. 34, August 20, 1990, pp. 29, 34.
49. Brooks, Jr., The Mythical Man-Month, Addison Wesley Publishing Company, Reading, Massachusetts, 1975.
50. Brown, Object-Oriented Databases: Applications in Software Engineering, McGraw-Hill, New York, New York, 1991.
51. Canning. "Progress in Project Management," EDP Analyzer, December 1977, pp. 1 - 11.
52. Carbato and C. Clingen. "A Managerial View of the Multics System Development." in Research Directions in Software Technology, Edited by P. Wegner, The M.I.T. Press, Cambridge, Massachusetts, 1979.
53. Cardenas and D. McLeod, Editors, Research Foundations in Object-Oriented and Semantic Database Systems, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
54. Coad and E. Yourdon, OOA -- Object-Oriented Analysis, 2nd Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
55. Coad and E. Yourdon, Object-Oriented Design, Prentice Hall, Englewood Cliffs, New Jersey, 1991.

56. Coad, "Why Use Object-Oriented Development (A Management Perspective)," *Journal of Object-Oriented Programming*, Vol. 4, No. 6, October 1991, pp. 60 - 61.
57. Connell, "Adopting Objects: a Path," *The Hotline on Object-Oriented Technology*, Vol. 3, No. 2, December 1991, pp. 5 - 8.
58. Cook, *ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming*, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989.
59. Canning. "Progress in Project Management," *EDP Analyzer*, December 1977, pp. 1 - 11.
60. Carbatto and C. Clingen. "A Managerial View of the Multics System Development." in *Research Directions in Software Technology*, Edited by P. Wegner, The M.I.T. Press, Cambridge, Massachusetts, 1979.
61. Cardenas and D. McLeod, Editors, *Research Foundations in Object-Oriented and Semantic Database Systems*, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
62. Coad and E. Yourdon, *OOA -- Object-Oriented Analysis*, 2nd Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
63. Coad and E. Yourdon, *Object-Oriented Design*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
64. Coad, "Why Use Object-Oriented Development (A Management Perspective)," *Journal of Object-Oriented Programming*, Vol. 4, No. 6, October 1991, pp. 60 - 61.
65. Connell, "Adopting Objects: a Path," *The Hotline on Object-Oriented Technology*, Vol. 3, No. 2, December 1991, pp. 5 - 8.
66. Cook, *ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming*, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989.
67. Dittrich, U. Dayal, and A.P. Buchmann, Editors, *On Object-Oriented Database Systems*, Springer-Verlag, New York, New York, 1991.
68. Dittrich, Editor, *Advances in Object-Oriented Database Systems*, Springer-Verlag, New York, New York, 1989.
69. Duhl and C. Damon, "A Performance Comparison of Object and Relational Databases Using the Sun Benchmark," *OOPSLA '88 Conference Proceedings*, Special Issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 153 - 163.
70. Dunn, *Software Quality Concepts and Plans*. Prentice Hall, Englewood Cliffs, New Jersey, 1990.
71. Ernst and Young, *Navigator Systems Series Overview Monograph*, Ernst and Young, 1990.
72. Foreman and J. Goodenough, *Ada Adoption Handbook: A Program Manager's Guide Version 1.0 Report CMU/SEI-87-TR--110*, Carnegie-Mellon University Software Engineering Institute, Pittsburgh, Pennsylvania, 1987.

73. Freedman and G.M. Weinberg, Handbook of Walkthroughs, Inspections, and Technical Reviews, 3rd Edition, Dorset House Publishing, New York, New York, 1990.
74. Gause and G.M. Weinberg, Are Your Lights On?, Dorset House Publishing, New York, New York, 1990.
75. Gjessing and K. Nygaard, ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988.
76. Glass, Software Conflict - Essays on the Art and Science of Software Engineering, Yourdon Press/Prentice Hall, Englewood Cliffs, New Jersey, 1990.
77. Goldberg and D. Robson, Smalltalk-80: The Language and Its Implementation, Addison-Wesley, Reading, Massachusetts, 1983.
78. Goldberg and K. Rubin, "Taming Object-Oriented Technology," Computer Language Magazine, Vol. 7, No. 10, p. 34 - 45, October, 1990.
79. Goldberg, "Issues in Managing Object-Oriented Projects," Presentation of the 5th Autumn School, September 11, 1991.
80. Gunther, Management Methodology for Software Product Engineering, John Wiley and Sons, New York, New York, 1978.
81. Gupta and E. Horowitz, Editors, Object-Oriented Databases With Applications to CASE, Networks, and VLSI CAD, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
82. Hanna, "Beyond the Waterfall Lies a Brave New World," Software Magazine, Vol. 11, No. 7, June 1991, p. 44 - 60.
83. Heitz, "HOOD: A Hierarchical Object-Oriented Design Method," Proceedings of the Third German Ada Users Congress, January 1988, Gesellschaft für Software Engineering, Munich, West Germany, pp. 12-1 - 12-9.
84. Henderson-Sellers and J.M. Edwards, "The Object-Oriented Systems Life-Cycle," Communications of the ACM, Vol. 33, No. 9, September, 1990, pp. 145 - 149.
85. Henderson-Sellers, A Book of Object-Oriented Knowledge, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
86. Hollocker, Software Reviews and Audits Handbook, John Wiley and Sons, New York, New York, 1990.
87. Hornung, "Transition to AdaBIS," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 289 - 298.
88. House, The Human Side of Project Management. Addison-Wesley Publishing Company, Reading, Massachusetts, 1988.
89. Hanna, "Beyond the Waterfall Lies a Brave New World," Software Magazine, Vol. 11, No. 7, June 1991, p. 44 - 60.

90. Heitz, "HOOD: A Hierarchical Object-Oriented Design Method," Proceedings of the Third German Ada Users Congress, January 1988, Gesellschaft für Software Engineering, Munich, West Germany, pp. 12-1 - 12-9.
91. Henderson-Sellers and J.M. Edwards, "The Object-Oriented Systems Life-Cycle," Communications of the ACM, Vol. 33, No. 9, September, 1990, pp. 145 - 149.
92. Henderson-Sellers, A Book of Object-Oriented Knowledge, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
93. Hollocker, Software Reviews and Audits Handbook, John Wiley and Sons, New York, New York, 1990.
94. Hornung, "Transition to AdaBIS," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 289 - 298.
95. House, The Human Side of Project Management. Addison-Wesley Publishing Company, Reading, Massachusetts, 1988.
96. Hanna, "Beyond the Waterfall Lies a Brave New World," Software Magazine, Vol. 11, No. 7, June 1991, p. 44 - 60.
97. Heitz, "HOOD: A Hierarchical Object-Oriented Design Method," Proceedings of the Third German Ada Users Congress, January 1988, Gesellschaft für Software Engineering, Munich, West Germany, pp. 12-1 - 12-9.
98. Henderson-Sellers and J.M. Edwards, "The Object-Oriented Systems Life-Cycle," Communications of the ACM, Vol. 33, No. 9, September, 1990, pp. 145 - 149.
99. Henderson-Sellers, A Book of Object-Oriented Knowledge, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
100. Hollocker, Software Reviews and Audits Handbook, John Wiley and Sons, New York, New York, 1990.
101. Hornung, "Transition to AdaBIS," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 289 - 298.
102. House, The Human Side of Project Management. Addison-Wesley Publishing Company, Reading, Massachusetts, 1988.
103. Kim, J.-M. Nicolas, and S. Nishio, Editors, Deductive and Object-Oriented Database: Proceedings of the First International Conference on Deductive and Object-Oriented Databases (DOOD 89) KyotoResearch Park, Kyoto, Japan, 4-6 December, 1989, North-Holland (Elsevier), New York, New York, 1990.
104. Kim, Introduction to Object-Oriented Databases, The MIT Press, Cambridge, Massachusetts, 1990.
105. King, Project Management Made Simple, Yourdon Press/Prentice Hall, Englewood Cliffs, New Jersey, 1992.
106. Kleim, "Project Proficiency," ComputerWorld, Vol. 29, No. 29, July 22, 1991, pp. 77 - 78.

107. Korson, V. Vaishnavi, and B. Meyer, Editors, Technology of Object-Oriented Languages and Systems: Tools 5, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
108. Lieberherr and I.M. Holland, "Assuring Good Style for Object-Oriented Programs," IEEE Software, Vol. 6, No. 5, September 1989, pp. 38 - 48.
109. Lieberherr and A.J. Riel, "Demeter: a CASE Study of Software Growth Through Parameterized Classes," Journal of Object-Oriented Programming, Vol. 1, No. 3, August/September 1988, pp. 8 - 22.
110. Lorenz, "Getting Started With Object Technology: Effectively Planning For Change," The Hotline on Object-Oriented Technology, Vol. 2, No. 11, September 1991, pp. 9 - 12.
111. McQuown, "Object-Oriented Design in a Real-Time Multiprocessor Environment," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 570 - 588.
112. Meyer, Object-Oriented Software Construction, Prentice Hall, Englewood Cliffs, New Jersey, 1988.
113. Millikin, "Object Orientation: What It Can Do For You," ComputerWorld, Vol. 23, No. 11, March 13, 1989, pp. 103 - 113.
114. Myers, Advances in Computer Architecture, Second Edition, John Wiley & Sons, New York, New York, 1982.
115. Nahouraii and F. Petry, Editors, IEEE Tutorial on Object-Oriented Databases, IEEE Catalog Number EH0332-7, IEEE Computer Society Press, Los Alamitos, California, 1991.
116. Norman, The Psychology of Everyday Things, Basic Books, Inc., New York, New York, 1988.
117. Organick, A Programmer's View of the Intel 432 System, McGraw-Hill, New York, New York, 1983.
118. Ould, Strategies for Software Engineering, John Wiley and Sons., New York, New York, 1990.
119. Palmer, "Playing with STELLA," The American Programmer, New York, New York, Vol. 3, No. 9, September, 1990, pp. 29 - 36.
120. Paulk, B. Curtis, and M.B. Chrissis, Capability Maturity Model for Software, Report CMU-91-TR-24, The Software Engineering Institute, Carnegie-Mellon University, Pittsburgh, Pennsylvania, 1991.
121. Peterson, Tutorial: Object-Oriented Computing, Volume 1: Concepts, IEEE Catalog Number EH0257-6, IEEE Computer Society Press, Washington, D.C., 1987.
122. Peterson, Tutorial: Object-Oriented Computing, Volume 2: Implementations, IEEE Catalog Number EH0257-6, IEEE Computer Society Press, Washington, D.C., 1987.

123. Pountain, "Rekursiv: An Object-Oriented CPU," Byte, Vol. 13, No. 11, November 1988, pp. 341 - 349.
124. Rakos, Software Project Management for Small to Medium Sized Projects, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
125. Richardson, R. Schultz, and E. Berard, Object-Oriented Modeling and Design, Berard Software Engineering, Inc., Gaithersburg, Maryland, 1992.
126. Rochester, Editor, The Personnel Crunch: Perspectives on Information Management, John Wiley and Sons, Inc., New York, New York, 1982.
127. Roetzheim, Developing Software to Government Standards, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
128. Rosenau, Project Management for Engineers, Van Nostrand Reinhold Company, Inc., New York, New York, 1984.
129. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen, Object-Oriented Modeling and Design, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
130. Salmons and T.T. Babitsky, Editors, 1990 International OOP Directory, COOT, Inc., New York, New York, 1990.
131. The Software Engineering Institute, SEI Overview, Carnegie-Mellon University Software Engineering Institute, Pittsburgh, Pennsylvania, 1987.
132. Shlaer and S.J. Mellor, Object-Oriented Systems Analysis: Modeling the World In Data, Yourdon Press: Prentice Hall, Englewood Cliffs, New Jersey, 1988.
133. Shlaer and S.J. Mellor, Object-Oriented Systems Analysis: Modeling the World In States, Yourdon Press: Prentice Hall, Englewood Cliffs, New Jersey, 1992.
134. Shriver and P. Wegner, Editors, Research Directions in Object-Oriented Programming, The MIT Press, Cambridge, Massachusetts, 1987.
135. Stark and E.V. Seidewitz, "Towards a General Object-Oriented Ada Life-Cycle," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 213 - 222.
136. M.K Stewart, "Object Projects: What Can Go Wrong," Hotline on Object-Oriented Technology, Vol. 2, No. 6, April 1991, pp. 15 - 17.
137. Stoecklin, E.J. Adams, and S. Smith, "Object-Oriented Analysis," Proceedings of the Fifth Washington Ada Symposium, June 27 - 30, 1988, Association for Computing Machinery, New York, New York, 1988, pp. 133 - 138.
138. Stroustrup, The C++ Programming Language, Second Edition, Addison-Wesley, Reading, Massachusetts, 1991.
139. Tanniru, et al, "Causes of Turnover Among DP Professionals," Proceedings of the 8th Annual Computer Personnel Research Conference, Miami, Florida, June 1981.

140. Thayer, "Software Engineering Project Management, A Top-Down View," Tutorial: Software Engineering Project Management, IEEE Computer Science Press., Los Alamitos, California, 1988.
141. van Genuchten, "Why is Software Late? An Empirical Study of Reasons For Delay in Software Development," IEEE Transactions on Software Engineering, Vol. 17, No. 6, June 1991, p. 582 - 590.
142. Villard and J. Renerts, Interface, Lecture Notes, Presentation to Data Processing Staff at People's Drug, Alexandria, Virginia, February, 1983.
143. Wasserman, P. Pircher, and R.J. Muller, "An Object-Oriented Design Notation for Software Design Representation," IEEE Computer, Vol. 23, No. 3, March 1990, pp. 50 - 63.
144. Wasserman, "Object-Oriented Thinking," Object Magazine, September/October 1991, pp. 10 - 13.
145. Weber, M.C. Paulk, C.J. Wise, and J.V. Withey, Key Practices of the Capability Maturity Model, Report CMU-91-TR-25, The Software Engineering Institute, Pittsburgh, Pennsylvania, 1991.
146. Winblad, S.D. Edwards, and D.R. King, Object-Oriented Software, Addison-Wesley Publishing Company, Reading, Massachusetts, 1990.
147. Wirfs-Brock, B. Wilkerson, and L. Wiener, Designing Object-Oriented Software, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
148. Yourdon, Managing the System Life Cycle, 2nd Edition, Yourdon Press/Prentice Hall, Englewood Cliffs, New Jersey, 1988.
149. Zdonik and D. Maier, Editors, Readings in Object-Oriented Database Systems, Morgan Kaufmann Publishers, Inc. San Mateo, California, 1990.

MCCABES CYCLOMATIC NUMBER

1. Brauns, E.: Analysis of Software Complexity for Different Programming Languages (german). Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, October 1990, pp. 1-14.
2. Hansen, W.J.: Measurement of Program Complexity by the Pair. SIGPLAN Notices, 13(1978)3, pp. 29-33.
3. Maes, R.: A composed program complexity measure. Angewandte Informatik, 27(1985)1, pp. 9-16.
4. McCabe, T.J.: A Complexity Measure. IEEE Transactions on Software Engineering, 2(1976)4, pp. 308-320.
5. Myers, G.J.: An Extension to the Cyclomatic Measure of Program Complexity. SIGPLAN Notices, 12(1977)10, pp. 61-64.
6. Henderson-Sellers, B.; Tegarden, D.: The application of cyclomatic complexity to multiple entry/exit modules. Information Technology Research Centre Report No. 93/2, University of New South Wales, April 1993.

MEASUREMENT FOUNDATION

1. Baker, A.L.; Bieman, J.M.; Fenton, N.; Gustavson, D.A.; Melton, A.; Whitty, R.: A Philosophy for Software Measurement. *The Journal of Systems and Software*, 12 (1990), pp. 277-281.
2. Basili, V.R.: Recent Advances in Software Measurement. *Proceedings of the 12th International Conference on Software Engineering*, March~26-30, Nice, France, 1990, pp. 44-49.
3. Bevan, N.; Macloed, M.: Usability Assessment and Measurement in Kelly, M.: *Management and Measurement of Software Quality*, UNICOM SEMINARS, Middlesex, UK, 1993, pp. 167-192.
4. Bieman, J.M.; Fenton, N.E.; Gustafson, D.A.; Melton, A.; Whitty, R.: Moving from Philosophy to Practice in Software Measurement. in: Denvir et. al.: *Formal Aspects of Measurement*, Springer Verlag, 1992, pp. 38-59.
5. Bourque, P.; Cot'e, V.: An Experiment in Software Sizing with Structured Analysis Metrics. *The Journal of Systems and Software*, 12 (1990), pp. 159-172.
6. Curtis, B.: Measurement and Experimentation in Software Engineering. *Proceedings of the IEEE*, 68(1980)9, pp. 1144-1157.
7. Curtis, B.: Software Metrics: Guest Editor's Introduction. *IEEE Transactions on Software Engineering*, 9(1983)6, pp. 637-638.
8. Ejiogu, L.O.: Beyond Structured Programming: An Introduction to the Principles of Applied Software Metrics. *Structured Programming*, (1990) 11, pp. 27-43.
9. Fenton, N.E.: Software Measurement: A necessary scientific basis. submitted to the *IEEE Transactions on Software Engineering*, July 1992, (31 p.).
10. Fenton, N.E.: Software Measurement: Why a Formal Approach? in: Denvir et. al.: *Formal Aspects of Measurement*, Springer Verlag, 1992, pp. 3-27.
11. Fenton, N.E.: The Mathematics of Complexity in Computing and Software Engineering. in: Johnson; Loomes: *The Mathematical Revolution Inspired by Computing*. Oxford University Press 1991, pp. 243-256.
12. Fenton, N.E.: When a software measure is not a measure. *Software Engineering Journal*, Sept. 1992, pp~357-362.
13. Fenton, N.E.; Hill, G.: *Systems Construction and Analysis - A Mathematical and Logical Framework*. McGraw-Hill Int., 1993.
14. Fenton, N.E.; Kaposi, A.A.: Metrics and Software Structure. *Information and Software Technology*, 29(1987)6, pp. 301-320.
15. Fenton, N.; Melton, A.: Deriving Structurally Based Software Measures. *The Journal of Systems and Software*, 12 (1990), pp. 177-187.
16. Fenton, N.E.; Whitty, R.W.: Axiomatic approach to Software Metrication through Program Decomposition. *The Computer Journal*, 29(1986)4, pp. 330-339.

17. Finkelstein, L.; Leaning, M.S.: A review of the fundamental concepts of measurement. *Measurement*, 2(1984)1, pp. 25-34.
18. Harrison, W.: *Software Metrics and Decision-Making The Software QA*, Beaverton, OR, 1(1994)1, pp. 5-12.
19. Hetzel, B.: The Sorry State of Software Practice Measurement and Evaluation. *Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry*, Amsterdam, 29 September - 1 October 1993, Section 16.
20. Jones, C.: *Critical Problems in Software Measurement*. Research Report, SPR Inc., Nov. 1992 (64 p.).
21. Kearney, J.K.; Sedlmeyer, R.L.; Thompson, W.B.; Gray, M.A.; Adler, M.A.: *Software Complexity Measurement*. *Comm. of the ACM*, 29(1986)11, pp. 1044-1050.
22. Kitchenham, B.: Never Mind the Metrics What About the Numbers! in: Denvir et. al.: *Formal Aspects of Measurement*, Springer Verlag, 1992, pp. 28-37.
23. Krantz, D.H. et al.: *Foundations of Measurement*. Academic Press, 1971.
24. Miluk, G.: The Role of Measurement in Software Process Improvement. *Proceedings of the Third International Conference on Software Quality*, Lake Tahoe, Nevada, 4-6 October 1993, pp. 191-197.
25. Pfleeger, S.L.: *Experimental Design and Analysis in Software Engineering*. *Software Engineering Notes*, 20(1995)1, pp. 23-26.
26. Poore, J.H.: *Derivation of Local Software Quality Metrics (Software Quality Circles)*. *Software -- Practice and Experience*, 18(1988)11, pp. 1017-1027.
27. Poppe, S.: *Application of the calibrating method to the software measurement (german)*. Study, TU Magdeburg, July 1992.
28. Roberts, F.S.: *Measurement Theory with Applications to Decisionmaking, Utility, and the Social Science*. Addison-Wesley Publ. Comp., 1979.
29. Roche, J.; Jackson, M.; Shepperd, M.: *Software Measurement Methods: An Evaluation and Perspective*. *Proc. of the Third Symposium on Assessment of Quality Software Development Tools*, Washington D.C., June 7-9, 1994, IEEE Computer Society Press, pp. 0-69.
30. Rombach, H.D.; Basili, V.R.; Selby, R.W.: *Experimental Software Engineering Issues: Critical Assessment and Future Directions*. Springer Publisher, *Lecture Notes in Computer Science* 706, 1993.
31. Russell, M.: *The Mathematics of Measurement in Software Engineering*. in Denvir et. al.: *Formal Aspects of Measurement*. Springer Verlag, 1992, pp. 209-218.
32. Schaefer, H.: *How to assure the survival of software metrics data collection*. in: Dumke/Zuse: *Theorie und Praxis der Softwaremessung*, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp. 45-58.

33. Schneidewind, N.F.: New Software-Quality Metrics Methodology Standard fills measurement need. *IEEE Computer*, April 1993, pp. 105-106.
34. Shah-Jarvis, A.: Improvement Process Based on Metrics. *Proceedings of the Third International Conference on Software Quality*, Lake Tahoe, Nevada, 4-6 October 1993, pp. 93-100.
35. Sheil, B.A.: The Psychological Study of Programming. *Computing Surveys*, 13(1981)1, pp. 102-120.
36. Smith, E.: Comparability Orders and Measurement. GMD paper 642, Sankt Augustin, (36 p.), April 1992.
37. Zuse, H.: How to Cope with Software Complexity *Proceedings of the International Conference on Practical Improvement of Software Processes and Products*, 4-5 May, 1994, Dublin, Ireland.
38. Zuse, H.; Bollmann-Sdorra, P.: Measurement Theory and Software Measures. in: Denvir et. al.: *Formal Aspects of Measurement*. Springer Verlag, 1992, pp. 219-259.

MEASUREMENT FRAMEWORKS

1. Abreu, F.B.e; Carapuca, R.: Candidate Metrics for Object-Oriented Software within a Taxonomy Framework. *Journal of Systems and Software*, 26(1994)1, pp. 87-96.
2. Bache, R.; Neil, M.: Introducing Metrics into Industry: A Perspective on GQM. *Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry*, Amsterdam, 29.9.-1.10.1993, Section 26.
3. Basili, V.R.: Applying the Goal Question Metric Paradigm in the Experience Factory. *Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry*, Amsterdam, 29.9.-1.10.1993, Section 1.
4. Basili, V.R.; Selby, R.W.; Hutchens, D.H.: Experimentation in Software Engineering. *IEEE Transactions on Software Engineering*, 12(1986)7, pp. 733-743.
5. Benford, S.D.; Burke, E.K.; Foxley, E.: Integrating Software Quality Assurance Into the Teaching of Programming. *Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry*, Amsterdam, 29.9.-1.10.1993, Section 23.
6. Bhide, S.: Generalized Software Process-integrated Metrics Framework. *The Journal of Systems and Software*, 12 (1990), pp. 249-254.
7. Biffl, S.; Grechening, T.: Establishing reviewing and automatic analysis fo code in everyday's coding practice. *Proceedings of the Third International Conference on Software Quality*, Lake Tahoe, Nevada, 4-6 October 1993, pp. 165-172.
8. Bush, M.E.; Fenton, N.E.: Software Measurement: A Conceptual Framework. *The Journal of Systems and Software*, 12 (1990), pp. 223-231.
9. Bush, M.; Russell, M.: A New Modular Course For Teaching About Software Engineering Measurement Within Academia. Project No. 2384 - METKIT, London, 1991.

10. Cavano, J.P.: Toward High Confidence Software. IEEE Transactions on Software Engineering, 11(1985)12, pp.~1449-1455.
11. Cole, R.J.; Woods, D.: Measurement Through The Software Lifecycle: A Comparative Case Study. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29.9.-1.10.1993, Section 19.
12. Cox, G.M.: Sustaining a Software Metrics Programme in Industry. in: Fenton/Littlewood: Software Reliability and Metrics, Elsevier Applied Science, London New York, 1991, pp. 1-15.
13. Daskalantonakis, M.K.: A Practical View of Software Measurement and Implementation Experiences Within Motorola. IEEE Transactions on Software Engineering, 18(1992)11, pp. 998-1010.
14. Debou, C.; Pescoller, L.; Fuchs, N.: Software measurements on telecom sytems - Success stories? Proceedings of the Third European Conference on Software Quality, Madrid, November 1993.
15. DeMarco, T.: Controlling Software Projects - Management Measurement & Estimation. Yourdon Inc./ Prentice-Hall Inc., 1982.
16. Dumke, R.; Goetzke, C.; Winkler, A.: Total Quality Management In Object-Oriented Software Development. Proceedings of the Third International Confenrence on Software Quality, Lake Tahoe, Nevada, 4-6 October 1993, pp. 109-116.
17. Dumke, R.; Kuhrau, I.: Tool-Based Quality Management in Object-Oriented Software Development. Proc. of the Third Symposium on Assessment of Quality Software Development Tools, Washington D.C., June 7-9, 1994, IEEE Society Press, pp. 148-160.
18. Ebert, C.: A Method for controlling the complexity in the software development process. (german) Ph.D., VDI Publisher, Duesseldorf, 1995.
19. Espley, R.: Industrial Experience - Working With AMI. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29.9.-1.10.1993, Section 8.
20. Evanco, W.M.; Lacovara, R.: A Model-Based Framework for the Integration of Software Metrics. The Journal of Systems and Software, 26(1994)1, pp. 77-86.
21. Fuchs, N.: Software Measurement -- An Evolutionary Approach. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29.9.-1.10.1993, Section 9.
22. Gilchrist, T.: A Framework For Measurement Programs. Proceedings of the Third International Confenrence on Software Quality, Lake Tahoe, Nevada, 4-6 October 1993, pp. 101-108.
23. Goodman, P.: The Practica Implementation of Process Improvement Initiatives. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29.9.-1.10.1993, Section 20.
24. Grady, R.B.: Practical Software Metrics for Project Management and Process Improvement. Prentice Hall, 1992.

25. Grady, R.B.; Caswell, D.L.: Software Metrics: Establishing a Company-Wide Program. Prentice-Hall Inc., New Jersey, 1987.
26. Jeffery, R.; Berry, M.: A Framework for Evaluation and Prediction of Metrics Program Success. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 28-39.
27. Kitchenham, B.A.; Linkman, S.G.; Law, D.T.: Critical review of quantitative assessment. Software Engineering Journal, 9(1994)2, pp.43-53.
28. Knight, J.C.; Myers, E.A.: An Improved Inspection Technique. Comm. of the ACM, 36(1993)11, pp. 51-61.
29. Koch, G.R.: Beyond SEI's CMM -- The European BOOTSTRAP Approach. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29.9.-1.10.1993, Section 15.
30. Liggesmeyer, P.: Software Complexity Metrics as Basis for choosing test methods. (german) Softwaretechnik-Trends, 13(1993)3, pp. 103-110.
31. Lytz, R.: Software Metrics for the Boeing 777. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon.
32. Magee, P.: Starter Metrics System for Small Companies. Proc. of the 11th CSR Conference on Software Evolution, Models and Metrics. September 7-9, Dublin, Ireland, Section 5.
33. Matsumoto, K.; Kusumoto, S.; Kikuno, T.; Torii, K.: A New Framework of Measuring Software Development Process. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 108-118.
34. Miluk, G.: Software Kinetics -- Developing a Process for Software Process Improvement. Proceedings of the International Software Quality Conference, Dayton, Ohio, 1991, pp. 56-65.
35. Morell, L.J.; Voas, J.M.: A Framework for Defining Semantic Metrics. The Journal of Systems and Software, 20(1993)3, pp. 245-251.
36. Muellerburg, M.; Meyerhoff, D.; Flacke, S.: Enhancing Accessibility of Metrics Knowledge. Proceedings of the EUROMETRICS'91, March 1991.
37. Myers, G.J.: A Controlled Experiment in Program Testing and Code Walkthroughs/Inspections. Comm. of the ACM, 21(1978)9, pp. 760-768.
38. Perkins, R.L.: The Characteristics Based Metrics Model: A Disciplined Approach to Implementing a Software Metrics Methodology. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 34-39.
39. Pfleeger, S.L.: A metrics program to address corporate needs should include linking metrics to process maturity, a tools-evaluation database, and the use of multiple-metrics graphs. IEEE Software, May 1993, pp. 67-74.

40. Pfleeger, S.L.: Setting Up a Metrics Program in Industry. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29 September - 1 October 1993, Section 4.
41. Pfleeger, S.L.; McGowan, C.: Software Metrics in the Process Maturity Framework. The Journal of Systems and Software, 12 (1990), pp. 255-261.
42. Redman, T.C.: Data Quality -- Management and Technology. Bantam Books, New York London Sydney, 1992.
43. Robillard, P.N.; Coupal, D.; Coallier, F.: Profiling Software Through the Use of Metrics. Quality Engineering Workshop, Ottawa, 16.-17. Oktober, 1991.
44. Rombach, H.D.: Practical Benefits of Goal-Oriented Measurement. in: Fenton/Littlewood: Software Reliability and Metrics, Elsevier Applied Science, London New York, 1991, pp. 217-235.
45. Rombach, H.D.; Basili, V.R.; Selby, R.W.: Experimental Software Engineering Issues: Critical Assessment and Future Directions. Proc. of the International Workshop in Dagstuhl Castl, September 1992, Lecture Notes in Computer Science 706, Springer Publ., Berlin Heidelberg New York, 1993.
46. Rudolf, E.: Experience in software metrics programs in practice. (german) in: Dumke/Zuse: Theorie und Praxis der Softwaremessung, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp. 35-44.
47. Russell, M.: International Survey of Software Measurement Education and Training. The Journal of Systems and Software, 12 (1990), pp. 233-241.
48. Schneidewind, N.F.: Minimizing Risk in Applying Metrics on Multiple Projects. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, pp. 173-182.
49. Schulenklopper, J.; Vliet, J.C.v.; Swede, V.v.: Situational Metrics. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29.9.-1.10.1993, Section 24.
50. Sunazuka, T.; Azuma, M.; Yamagishi, N.: Software Quality Assessment Technology. Proceedings of the 8th International Conference on Software Engineering, August 28-30, London, 1985, pp. 142-148.
51. Werner, A.: A practical view to the evaluation of software quality (german). Angewandte Informatik, 25(1983)6, pp. 242-251.
52. Whitty, R.; Bush, M.; Russell, M.: METKIT and the ESPRIT Program. The Journal of Systems and Software, 12 (1990), pp. 219-221.
53. Woodcock, T.G.; Koshgoftaar, T.M.: Metric Gathering Pitfalls. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon Yourdon, E.: The Decline and Fall of the American Programmer. Prentice-Hall Inc. 1992.

METRIC ANALYSIS

1. Basili, V.R.; Rombach, H.D.: The TAME Project: Towards Improvement-Oriented Software Environments. IEEE Transactions on Software Engineering, 14(1988)6, pp. 758-773.
2. Bush, M.; Kelly, M.: METKIT -- Metrics Educational Toolkit. London, 1991.
3. Conradt, O.; Holz, T.: A program to the determination of the McCabe complexity for a class of programs (german). Study, TU Magdeburg, 1990.
4. Deicke, P.; Krogel, M.: A program to the determination of the McCabe complexity with a choice of syntax (german). Study, TU Magdeburg, 1990.
5. Dumke, R.: Evaluated Software Processors. Research Report, TU Magdeburg, July 1989.
6. Harrison, W.: MAE: A Syntactic Metric Analysis Environment. The Journal of Systems and Software, (1988)8, pp. 57-62.
7. Hegewald, H.; Seibt, A.: A Program for Software Evaluation (german). Study, TU Magdeburg, 1988.
8. Illner, S.; Muenster, R.: A program for value determination of the Stetter metric for programs in Turbo pascal (german). Study, TU Magdeburg, 1990.
9. Matschos, M.; Neumann, K.: A program for the determination of the complexity of Smalltalk methods (german). Study, TU Magdeburg, 1991.
10. Matwin, S.; Missala, M.: A Simple, Machine Independent Tool for Obtaining Rough Measures of PASCAL Programs. SIGPLAN Notices, 11(1976)8, pp. 42-45.
11. Neumann, K.; Zimmermann, R.: An application of the Stetter metric to any programs (german). Study, TU Magdeburg, 1988.
12. Preuss, T.; Werthmann, T.: Computer aided determination of the Halstead measures for programs in Turbo Pascal (german). Study, TU Magdeburg, 1990.
13. Rendelmann, T.; Vaak, S.: Determination of the McCabe metric for syntactic independent source programs (german). Study, TU Magdeburg, 1990.
14. Schmitt, I.; Wettrau, D.: Computer Aided Determination of the Halstead measures with a optional syntax (german). Study, TU Magdeburg, 1990.
15. Sindermann, A.: Design and implementation of a prototype of a metric based compiler (german). Master's Thesis, TU Magdeburg, 1991.
16. Sneed, H.: PC-MESS -- a computer based tool for storage and evaluation of software metrics (german). SES Software-Engineering Service GmbH, Ottobrunn, Germany, 1992.
17. Tsalidis, C.: The ATHENA Software Measurement Tool. Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, October 1990, pp.~83-109.
18. Tsalidis, C.; Christodoulakis, D.: Composing Software Quality Tools with Software Quality Metrics. Second European Conference on Software Quality Assurance, Conf.Proc., Oslo, 1990.

19. Tsalidis, C.; Christodoulakis, D.; Meritsas, D.: ATHENA: A Software Measurement and Metrics Environment. *Software Maintenance: Research and Practice*, 4(1992), pp. 61-81.
20. Vernik, R.J.; Turner, I.: Techniques and Tools for Analysing Software Products. *Australian Computer Journal*, 24(1992)3, pp.~98-105.
21. Zuse, H.: METRICS. User Manual, Hawthorne, April 1988.

METRICS AND FORMAL GRAMMARS

1. Jacob, P.; Cahill, T.: Software Product Metrics as Attributes in an Attribute Grammar. *Proceedings of the Second International Conference on Software Quality*, Research Triangle Park, NC, October 5-7, pp. 40-49.
2. Whitty, R. W.: Multi-dimensional Software Metrics. in: Denvir et. al.: *Formal Aspects of Measurement*. Springer Verlag, 1992, pp. 116-141.

METRICS BIBLIOGRAPHY

1. ANSI/IEEE Std 729-1983: *Glossary of Software Engineering Terminology*, IEEE, New York, New York, 1984.
2. Albrecht and J.E. Gaffney, "Software Function, Source Lines of Code, and Development Effort Prediction," *IEEE Transactions on Software Engineering*, Vol. SE-9, No. 6, November 1983, pp. 639 - 647.
3. Arthur, *Measuring Programmer Productivity and Software Quality*, John Wiley and Sons, New York, New York, 1985.
4. Bailey, T.P. Frazier, and J.W. Bailey, *A Descriptive Evaluation of Automated Software Cost Estimation Models*, Institute for Defense Analysis Paper P-1979, Institute for Defense Analysis, Alexandria, Virginia, October 1986.
5. Baker, "Implementing an Initial Software Metrics Program," *IEEE Proceedings of the National Aerospace and Electronics Conference*, Vol. 3, pp. 1289 - 1294.
6. Baker, J. Bieman, N. Fenton, A. Melton, and R. Whitty, "A Philosophy of Software Measurement," *Journal of Systems and Software*, Vol. 12, No. 3, July 1990, pp. 277 - 281.
7. Barnes and B.R. Swim, "Inheriting Software Metrics," *Journal of Object-Oriented Programming*, Vol. 6, No. 7, November/December 1993, pp. 27 - 34.
8. R. Basili, Editor, *Tutorial on Models and Metrics for Software Management and Engineering*, IEEE Computer Society Press (catalog number EHO-167-7), New York, New York, 1980.
9. Basili and R. Reiter, "Evaluating Automatable Measures of Software Models," *IEEE Workshop on Quantitative Software Models*, Kiamesha, New York, 1979, pp. 107 - 116.
10. Basili and D.H. Hutchens, "An Empirical Study of a Syntactic Complexity Family," *IEEE Transactions on Software Engineering*, Vol. SE-9, No. 6, November 1983, pp. 663 - 672.

11. Basili, R.W. Selby, Jr. and T.-Y. Phillips, "Metric Analysis and Data Validation Across FORTRAN Projects," IEEE Transactions on Software Engineering, Vol. SE-9, No. 6, November 1983, pp. 652 - 663.
12. Baumert, Software Measures and the Capability Maturity Model, CMU/SEI-92-TR-25, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1992.
13. Beane, N. Giddings, and J. Silverman, "Quantifying Software Designs," Proceedings of the Seventh International Conference on Software Engineering, 1984, pp. 314 - 322.
14. Behrens, "Measuring the Productivity of Computer Systems Development," IEEE Transactions on Software Engineering, Vol. SE-9, No. 6, November 1983, pp. 648 - 651.
15. Berard, "Engineering Ada: Halstead's Metrics and Ada," Ada Letters, Vol. 3, No. 3, November/December 1983, pp. 33 - 44.
16. Berard, "Engineering Ada: Halstead's Metrics and Ada -- Part II," Ada Letters, Vol. 3, No. 5, March/April 1984, pp. 44 - 49.
17. Berard, Essays on Object-Oriented Software Engineering, Volume 1, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
18. Bieman, A. Baker, P. Clites, D. Gustafson, and A. Melton, "A Standard Representation of Imperative Language Programs for Data Collection and Software Measures Specification," Journal of Systems and Software, Vol. 8, No. 1, January 1988, pp. 13 - 37.
19. Bieman and J. Schultz, "Estimating the Number of Test Cases Required to Satisfy the All-Do-Paths Testing Criterion," Proceedings of the Software Testing and Verification Symposium, TAV3-SIGSOFT89, December 1989, pp. 179-186.
20. Bieman and L.M. Ott, "Measuring Functional Cohesion," IEEE Transactions on Software Engineering, Vol. 20, No. 8, August 1994, pp. 644 - 657.
21. Boehm, J.R. Brown and M. Lipow, "Quantitative Evaluation of Software Quality," Tutorial on Models and Metrics for Software Management and Engineering, ed. V.R. Basili, IEEE Computer Society Press, Silver Spring, Maryland, 1980, pp. 218 - 231.
22. W. Boehm, Software Engineering Economics, Prentice Hall, Englewood Cliffs, New Jersey, 1981.
23. Bouldin, "What Are You Measuring? Why Are You Measuring It?," Software Magazine, Vol. 9, No. 10, August 1989, pp. 30 - 39.
24. Brooks, Jr., "Essence and Accidence of Software Engineering," IEEE Computer, Vol. 30, No. 4, April 1987.
25. Calvano and J.A. McCall, "A Framework for the Measurement of Software Technology," Tutorial on Software Quality Assurance A Practical Approach, ed. T.S. Chow, IEEE Computer Society Press, New York, New York, 1985, pp. 46 - 52.
26. Card, G.T. Page, and F.E. McGarry, "Criteria for Software Modularization," Proceedings of the IEEE Eighth International Conference on Software Engineering, August 1985, pp. 372 - 377.

27. Card, V.E. Church, and W.W. Agresti, "An Empirical Study of Software Design Practices," IEEE Transactions on Software Engineering, Vol. 12, No. 2, February 1986, pp. 264 - 271.
28. Card and W.W. Agresti, "Measuring Software Design Complexity," Journal of Systems and Software, Vol. 8, pp. 185 - 197.
29. Card and R.L. Glass, Measuring Software Design Quality, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
30. Cherniavsky and C.H. Smith, "On Weyuker's Axioms for Software Complexity Measures," IEEE Transactions on Software Engineering, Vol. 17, No. 6, June 1991, pp. 636 - 638.
31. Chidamber and C.F. Kemerer, "Towards a Metrics Suite for Object-Oriented Design," OOPSLA '91 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 26, No. 11, November 1991, pp. 197 - 211.
32. Chidamber and C.F. Kemerer, "A Metrics Suite for Object-Oriented Design," IEEE Transactions on Software Engineering, Vol. 20, No. 6, June 1994, pp. 476 - 493.
33. Conte, H.E. Sunsmore, and V.Y. Shen, Software Engineering Metrics and Models, Benjamin/Cummings, Menlo Park, California, 1986.
34. Curtis, S.B. Sheppard and P. Milliman, "Third Time Charm: Stronger Prediction of Performance by Software Complexity Measures," Tutorial on Programming Productivity: Issues for the Eighties, Edited by C. Jones, IEEE Computer Society Press, New York, New York, 1981, pp. 81 - 85.
35. Dale and H. Van Der Zee, "Software Productivity Metrics -- Who Needs Them?", Eurometrics '92 Proceedings, April 1992, pp. 31 - 43.
36. Daskalantonakis, "A Practical View of Software Measurement and Implementation Experiences Within Motorola," IEEE Transactions on Software Engineering, Vol. 18, No. 11, November 1992, pp. 998 - 1010.
37. Dreger, Function Point Analysis, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
38. Duhl and C. Damon, "A Performance Comparison of Object and Relational Databases Using the Sun Benchmark," OOPSLA '88 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 153 - 163.
39. H. Dunn, Software Quality: Concepts and Plans, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
40. O. Ejiogu, "The Critical Issues of Software Metrics, Part 0: Perspectives on Software Metrics," SIGPLAN Notices, Vol. 22, No. 3, March 1987, pp. 59 - 64.
41. Ejiogu, Software Engineering With Formal Metrics, QED Technical Publishing Group, Boston, Massachusetts, 1991.
42. Fenton, Software Metrics: A Rigorous Approach, Chapman and Hall, New York, New York, 1991.

43. Fenton, "Software Measurement: A Necessary Scientific Basis," IEEE Transactions on Software Engineering, Vol. 20, No. 3, March 1994, pp. 199 - 206.
44. Fichman and C. Kemerer, "Object-Oriented and Conventional Analysis and Design Methodologies: Comparison and Critique," IEEE Computer, Vol. 25, No. 10, October 1992, pp. 20 - 39.
45. Gilb, Software Metrics, Winthrop Publishers, Inc., Cambridge, Massachusetts, 1977.
46. Gill and C.F. Kemerer, "Cyclomatic Complexity Density and Software Maintenance Productivity," IEEE Transactions on Software Engineering, Vol. 17, No. 12, December 1991, pp. 1284 - 1288.
47. Goodman, "Implementing Software Metrics Programmes: A Project-Based Approach," Eurometrics '92 Proceedings, April 1992, pp. 147 - 151.
48. Gotterbarn, "A Distributed Compilation Environment -- Lessons Learned," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 120 - 127.
49. Grady, Practical Software Metrics for Project Management and Process Improvement, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
50. Grady, "Successfully Applying Software Metrics," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 18 - 25.
51. Grady and D.L. Caswell, Software Metrics: Establishing a Company-Wide Program, Prentice Hall, Englewood Cliffs, New Jersey, 1987.
52. Halstead, Elements of Software Science, Elsevier, North-Holland, New York, 1977.
53. Harrison, K. Magel, R. Kluczny, and A. DeKock, "Applying Software Complexity Metrics to Program Maintenance," IEEE Computer, Vol. 15, No. 9, September 1982, pp. 65 - 79.
54. Henderson-Sellers, "Modularization and McCabe's Cyclomatic Complexity," Communications of the ACM, Vol. 35, No. 12, December 1992, pp. 17 - 19.
55. Henry and D. Kafura, "The Evaluation of Software Systems' Structure Using Quantitative Software Metrics," Software--Practice and Experience, Vol. 14, No. 6, June 1984, pp. 561 - 573.
56. Hetzel, Making Software Measurement Work: Building an Effective Measurement Program, QED Technical Publishing Group, Boston, Massachusetts, 1993.
57. Horgan, S. London, and M.R. Lyn, "Achieving Software Quality With Testing Coverage Measures," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 60 - 69.
58. Hufnagel and J.C. Brown, "Performance Properties of Vertically Partitioned Object-Oriented Systems," IEEE Transactions on Software Engineering, Vol. 15, No. 8, August 1989, pp. 935 - 946.
59. Jenson and J. Bartley, "Parametric Estimation of Programming Effort: An Object-Oriented Model," Journal of Systems and Software, Vol. 15, 1992, pp. 107 - 114.

60. Jones, "Measuring Programming Quality and Productivity," IBM Systems Journal, Vol. 17, No. 1, 1978, pp. 39 - 63.
61. Jones, Programming Productivity, McGraw-Hill Book Company, New York, New York, 1986.
62. Jones, Applied Software Measurement: Assuring Productivity and Quality, McGraw-Hill, Inc., New York, New York, 1991.
63. Kemerer and B.S. Porter, "Improving the Reliability of Function Point Measurement: An Empirical Study," IEEE Transactions on Software Engineering, Vol. 18, No. 11, November 1992, pp. 1011 - 1024.
64. Keyes, "New Metrics Needed for New Generation," Software Magazine, Vol. 12, No. 6, May 1992, pp. 42 - 56.
65. Khoshgoftaar and P. Oman, "Software Metrics: Charting the Course," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 13 - 15.
66. Kolewe, "Metrics in Object-Oriented Design and Programming," Software Development, Vol. 1, No. 4, October 1993, pp. 53 - 62.
67. Lanning and T.M. Khoshgoftaar, "Modeling the Relationship Between Source Code Complexity and Maintenance Difficulty," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 35 - 40.
68. Laranjeira, "Software Size Estimation of Object-Oriented Systems," IEEE Transactions on Software Engineering, Vol. 16, No. 5, May 1990, pp. 510 - 522.
69. Li and S. Henry, "Maintenance Metrics for the Object-Oriented Paradigm," Proceedings of the First International Software Metrics Symposium, Baltimore, Maryland, 1993, pp. 52 - 60.
70. Li, S. Henry and C. Selig, "Measuring Ada Design to Predict Maintainability," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 107 - 113.
71. Liberherr and I.M. Holland, "Assuring Good Style for Object-Oriented Programs," IEEE Software, Vol. 6, No. 5, September 1989, pp. 38 - 48.
72. Liberherr and A.J. Riel, "Demeter: a CASE Study of Software Growth Through Parameterized Classes," Journal of Object-Oriented Programming, Vol. 1, No. 3, August/September 1988, pp. 8 - 22.
73. Lohse and S.H. Zweben, "Experimental Evaluation of Software Design Principles: An Investigation Into the Effect of Module Coupling on System Modifiability," Journal of Systems and Software, Vol. 4, No. 4, November 1984, pp. 301 - 308.
74. Londeix, Cost Estimation for Software Development, Addison Wesley, Reading, Massachusetts, 1987.
75. Lorenz and J. Kidd, Object-Oriented Software Metrics, Prentice Hall, Englewood Cliffs, New Jersey, 1994.

76. McCabe, "A Complexity Measure," IEEE Transactions on Software Engineering, Vol. SE-2, No. 4, October 1976, pp. 243 - 245.
77. McCall, D. Markham, M. Stosick and R. McGindly, "The Automated Measurement of Software Quality," Tutorial on Software Quality Assurance A Practical Approach, ed. T.S. Chow, IEEE Computer Society Press, New York, New York, 1985, pp. 388 - 394.
78. A. Miller, "The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information," The Psychological Review, Vol. 63, No. 2, March 1956, pp. 81 - 97. Reprinted in K.H. Möller and D.J. Paulish, Software Metrics: A Practitioner's Guide to Improved Product Development, Chapman and Hall, London, United Kingdom, 1993.
79. J. Myers, Reliable Software through Composite Design, Van Nostrand Reinhold Company, New York, New York, 1975.
80. Paige, "A Metric for Software Test Planning," Tutorial on Software Quality Assurance Practical Approach, Edited by T.S. Chow, IEEE Computer Society Press, New York, New York, 1985, pp. 70-75.
81. Paulish and A.D. Carleton, "Case Studies of Software-Process-Improvement Measurement," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 50 - 57.
82. S.L Pfleeger, N. Fenton, and S. Page, "Evaluating Software Engineering Standards," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 71 - 79.
83. Pfleeger and J.D. Palmer, "Software Estimation for Object Oriented Systems," Fall International Function Point Users Group Conference, San Antonio, Texas, October 1-4, 1990, pp. 181 - 196.
84. Pfleeger and J.C. Fitzgerald, Jr., "A Software Metrics Database: Support for Analysis and Decision-Making," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 114 - 119.
85. Putnam and W. Myers, Measures for Excellence: Reliable Software on Time, Within Budget, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
86. Rains, "Function Points In an Ada Object-Oriented Design?", OOPS Messenger, Vol. 2, No. 4, October 1991, pp. 23 - 25.
87. St. Dennis, P. Stachour, E. Frankowski, E. Onuegbe, "Measurable Characteristics of Reusable Ada Software," Ada Letters, Vol. VI, No. 2, pp. 41 - 50.
88. Stark, R.C. Durst, and C.W. Vowell, "Using Metrics In Management Decision Making," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 42 - 48.
89. Symons, Software Sizing and Estimating: Mk II FPA, John Wiley and Sons, New York, New York, 1991.
90. Troy and S.H. Zweben, "Measuring the Quality of Structured Designs," Journal of Systems and Software, Vol. 2, No. 2, June 1981, pp. 113 - 120.
91. Weller, "Using Metrics to Manage Software Projects," IEEE Computer, Vol. 27, No. 9, September 1994, pp. 27 - 33.

92. Weyuker, "Evaluating Software Complexity Measures," IEEE Transactions on Software Engineering, Vol. 14, No. 9, September 1988, pp. 1357 - 1365.
93. Whitmire, "Object-Oriented Measurement of Software," The Encyclopedia of Software Engineering, Volume 2, J.J. Marciniak, Editor, John Wiley and Sons, New York, New York, 1994, pp. 737 - 739.
94. Wild, III, "Managing Class Coupling," Unix Review, Vol. 9, No. 10, October 1991, pp. 45 - 47.

METRICS VALIDATION

1. Abe, J.; Sakamura, K.; Aiso, H.: An Analysis of Software Project Failure. Proceedings of the 4th International Conference on Software Engineering, September 17-19, Munich, 1979, pp. 378-385.
2. Basili, V.R.; Perricone, B.T.: Software Errors and Complexity: An Empirical Investigation. Comm. of the ACM, 27(1984)1, pp. 42-52.
3. Berg, K.G. van den; Broek, P.M. van den; Peterson, G.M. van: Validation of Structure Metrics: A Case Study. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 92-99.
4. Davis, J.S.; LeBlanc, R.I.: A Study of the Applicability of Complexity Measures. IEEE Transactions on Software Engineering, 14(1988)9, pp. 1366-1372.
5. Ejiogu, L.O.: Five Principles for the Formal Validation of Models of Software Metrics. to be published in SIGPLAN Notices, 1993.
6. Gustafson, D.A.; Tan, J.T.; Weaver, P.: Software Measure Specification. Proc. of the SOGSOFT'93, December 7-10, Los Angeles, in: SIGSOFT Notes, 18(1993)5, pp. 163 168.
7. Gustafson, D.A.; Toledo, R.M.; Courtney, R.E.; Tamsamani, N.: A Critique of Validation/Verification Techniques for Software Development Measures. in Denvir et. al. : Formal Aspects of Measurement. Springer Verlag, 1992., pp. 145-156.
8. Itzfeldt, W.D.; Schmidt, M.; Timm, M.: Specification of methods for the validation of software quality measures (german). Angewandte Informatik, 26(1984)1, pp. 12-21.
9. Kafura, D.; Canning, J.: A Validation of Software Metrics Using Many Metrics and Two Resources. Proceedings of the 8th International Conference on Software Engineering, August 28-30, London, 1985, pp. 378-385.
10. Kemerer, C.F.: An Empirical Validation of Software Cost Estimation Models. Comm. of the ACM, 30(1987)5, pp. 416-4429.
11. Lind, R.K.; Vairavan, K.: An Experimental Investigation of Software Metrics and Their Relationship to Software Development Effort. IEEE Transactions on Software Engineering, 15(1989)5, pp. 649-653.
12. MacDonell, S.G.: Rigor in Software Complexity Measurement Experimentation. Journal of Systems and Software, (1991)16, pp. 141-149.

13. Navlakha, J.K.: A Survey of System Complexity Metrics. *The Computer Journal*, 39(1987)3, pp. 233-238.
14. Payne, J.E.: Experiences in Measuring Software Quality. *Proceedings of the Internationale Software Quality Conference*, Dayton, Ohio, 1991, pp. 164-168.
15. Samson et al.: The Relationship between Specification and Implementation Metrics. in: Kitchenham; Littlewood: *Measurement for Software Control and Assurance*. Elsevier Science Publisher Ltd, 1989, pp. 335-384.
16. Schneidewind, N.F.: Methodology For Validating Software Metrics. *IEEE Transactions on Software Engineering*, 18(1992)5, pp. 410-422.
17. Shepperd, M.: Algebraic Models and Metric Validation. in: Denvir et. al.: *Formal Aspects of Measurement*. Springer Verlag, 1992, pp. 157-175.
18. Shepperd, M.; Ince, D.C.: A Critique of Three Metrics. *The Journal of Systems and Software*, 26(1993)3, pp. 197-210.
19. Shepperd, M.; Ince, D.C.: *Derivation and Validation of Software Metrics*. Oxford Science Publications, 1993.
20. Zuse, H.: Criteria for Program Comprehension Derived from Software Complexity Metrics. *Proceedings of the International Conference on Program Comprehension*, Capri, July 8-9, 1993.
21. Zuse, H.: Foundations of the Validation of Object-Oriented Software Measures. in: Dumke/Zuse: *Theorie und Praxis der Softwaremessung*, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp. 136-214.
22. Zuse, H.: Foundations of Validation, Prediction, and Metrics. *Tutorial Proceedings of the Third International Conference on Software Quality*, Lake Tahoe, 4-6 October 1993, pp. 205-266.
23. Zuse, H.: Foundations of Validation, Prediction, and Software Measures. *Proceedings of the Annual Workshop on Software Metrics*, April 10-12, 1994, Silver Falls, Oregon.
24. Zuse, H.: Validation for Software Complexity Measures from a Measurement Theoretic View. *Proceedings of the IEEE-CS International Software Metrics Symposium*, May 21-22, 1993.

MODIFIABILITY MEASURES

1. Briand, L.C.; Basili, V.R.: A Classification for the Effective Management of Changes during the Maintenance Process. *Proceedings of the Conference on Software Maintenance*, Orlando, Nov. 9-12 1992, pp. 328-336.
2. Cantone, G.; Cimitile, A.; Carlini, U. de: Well-formed Conversion of Unstructured One-in/one-out Schemes for Complexity Measurement and Program Maintenance. *The Computer Journal*, 20(1986)4, pp. 322-329.
3. Harrison, W.; Magel, K.; Kluczny, R.; DeKock, A.: Applying Software Complexity Metrics to Program Maintenance. *IEEE Computer*, (1982)9, pp. 65-79.

4. Munson, J.B.: Software Maintainability: A Practical Concern for Life-Cycle Costs IEEE Computer, November 1981, pp. 103-109.
5. Ott, L.M.; Bieman, J.M.: Effects of Software Changes on Module Cohesion. Proceedings of the Conference on Software Maintenance, Orlando, Nov. 9-12 1992, pp. 345-353.
6. Poppe, S.: Maintenance of system software on the base of a maintenance metrics (german). Master's Thesis, Technical University of Magdeburg, 1993.
7. Sneed, H.M.: Sense, purpose, and supports to the dynamic program analysis (german). Angewandte Informatik, 25(1983)8, pp. 321-327.
8. Zuse, H.: Measuring Factors Contributing to Software Maintenance Complexity. Second International Conference on Software Quality, Research Triangle Park, NC, October 1992.

MODULARIZATION MEASUREMENT

1. Banker, R.D.; Datar, S.M.; Kemerer, C.F.; Zweig, D.: Software Complexity and Maintenance Costs. Comm. of the ACM, 36(1993)11, pp. 81-94.
2. Bieman, J.M.; Ott, L.M.: Measuring Functional Cohesion. IEEE Transactions on Software Engineering, 29(1994)8, pp. 644-657.
3. Briand, L.C.; Thomas, W.M.; Hetmanski, C.J.: Modeling and Managing Risk Early in Software Development. Proceedings of the 15th International Conference on Software Engineering, May 17-21, Baltimore, 1993, pp. 55-65.
4. Card, D.N.: Designing Software for Producibility. The Journal of Systems and Software, 17(1992), pp. 219-225.
5. Card, D.N.; Agresti, W.W.: Measuring Software Design Complexity. The Journal of Systems and Software, (1988)8, pp. 185-197.
6. Card, D.N.; Church, V.E.; Agresti, W.W.: An Empirical Study of Software Design Practices. IEEE Transactions on Software Engineering, 12(1986)2, pp. 264-271.
7. Card, D.N.; Page G.T.; McGarry, F.E.: Criteria for Software Modularization. Proceedings of the 8th International Conference on Software Engineering, August 28-30, London, 1985, pp. 372-377.
8. Dhama, H.: Quantitative Models of Cohesion and Coupling in Software. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 11994, Silver Falls, Oregon.
9. Emerson, T.J.: A Discriminant Metric for Module Cohesion. Proceedings of the 7th International Conference on Software Engineering, Orlando, Florida, Match 1984, pp. 294-303.
10. Goedicke, M.: Paradigms of modular system development. in: Mitchell; R.J.: Managing Complexity in Software Engineering. IEE Computing Series 17, London, 1990, pp. 1-20.
11. Keutgen, H.: A metric for evaluation of the modularization (german). Lecture Notes on Computer Science 50, Springer Publisher, Berlin Heidelberg New York, 1981.

12. Kitchenham, B.A.; Pickard, L.M.; Linkman, S.J.: An evaluation of some design metrics. *Software Engineering Journal*, January 1990, pp. 50-58.
13. Lakothia, A.: Rule-Based Approach to Computing Module Cohesion. *Proceedings of the 15th International Conference on Software Engineering*, May 17-21, Baltimore, 1993, pp. 35-44.
14. Lohse, J.B.; Zweben, S.H.: Experimental Evaluation of Software Design Principles: An Investigation into Effect of Module Coupling on System Modifiability. *The Journal of Systems and Software*, 4 (1984), pp. 301-308.
15. McCabe, T.J.; Butler, C.W.: Design Complexity Measurement and Testing. *Comm. of the ACM*, 32(1989)12, pp. 1415-1425.
16. Mueller, K.: The use of a CASE tool based structured method to a project realization and the metric based comparison with the conventional development (german). *Master's Thesis*, TU Magdeburg, 1993.
17. Munson, J.C.; Khoshgoftaar, T.M.: Applications of a Relative Complexity Metric for Software Project Management. *The Journal of Systems and Software*, 12 (1990), pp. 283-291.
18. Offutt, A.J.; Harrold, J.; Kolte, P.: A Software Metric System for Module Coupling. *The Journal of Systems and Software*, 20(1993)3, pp. 295-308.
19. Ott, L.M.; Thuss, J.J.: Slice Based Metrics for Estimating Cohesion. *Proceedings of the First International Software Metrics Symposium*, Baltimore, May 21-22, 1993, pp. 71-81.
20. Rising, L.; Calliss, F.W.: An Information-Hiding Metric. *The Journal of Systems and Software*, 29(1994)3, pp. 211-220.
21. Rising, L.; Calliss, F.W.: Problems with Determining Package Cohesion and Coupling. *Software -- Practice and Experience*, 22(1992)7, pp. 553-571.
22. Sellers, B.H.: Modularization and McCabe's Cyclomatic Complexity. *Comm. of the ACM*, 35(1992)12, pp. 17-19.
23. Smith, L.M.C.; Samadzadeh, M.H.: Measuring Complexity and Stability of WEB Programs. *Structured Programming* (1992)13, pp. 35-50.
24. Troy, D.A.; Zweben, S.H.: Measuring the Quality of Structured Design. *The Journal of Systems and Software*, (1981)2, pp. 113-120.
25. Van Verth, P.B.: A Program Complexity Model that Includes Procedures. *Buffalo*, 1987.
26. Van Verth, P.B.: Testing A Model of Program Quality. *SIGCSE Conference 1986*, Cincinnati, February 1986, pp. 163-170.
27. Yau, S.S.; Collofello, J.S.: Design Stability Measures for Software Maintenance. *IEEE Transactions on Software Engineering*, 11(1985)9, pp. 849-856.

NETWORK MEASUREMENT

1. Abrams, M.D.; Treu, S.: A Methodology for Interactive Computer Service Measurement. Comm. of the ACM, 20(1977)12, pp. 936-944.
2. Hall, N.R.; Preiser, S.: Combined Network Complexity Measures. IBM Journal of Research and Development, 28(1984)1, pp. 15-27.
3. Pilz, S.; Foltin, E.: Foundations to the development of test metrics for local area networks (german). Study, TU Magdeburg, July 1992.

OBJECT-ORIENTED DESIGN MEASURES

1. Chidamber, S.R.; Kemerer, C.F.: A Metrics Suite for Object Oriented Design. CISR Working Paper No. 249, M.I.T., Cambridge, February 1993.
2. Chidamber, S.R.; Kemerer, C.F.: A Metrics Suite for Object Oriented Design. IEEE Transactions on Software Engineering, 20(1994)6, pp. 476-493.
3. Davis, N.W.; Irving, M.; Lee, J.E.: The evolution of object-oriented design from concept to method. in: Mitchell; R.J.: Managing Complexity in Software Engineering. IEE Computing Series 17, London, 1990, pp. 21-50.
4. Dumke, R.; Zuse, H.: Software Metrics in Object-Oriented Software Development. in: Lehner, F.: Die Wartung von wissensbasierten Systemen. Haensel-Hohenhausen Publ., Frankfurt, Washington, 1994, pp. 58-96.
5. Dvorak, J.: Conceptual Entropy and Its Effects on Class Hierarchies. IEEE Computer, June 1994, pp. 59-63.
6. Henderson-Sellers, B.: A Book of Object-Oriented Knowledge. Prentice Hall Inc., 1992.
7. Henderson-Sellers, B.; Tegarden, D.; Monarchi, D.: Object-oriented Metrics. Tutorial on the ECOOP'93, 26 July 1993, Kaiserslautern, Germany.
8. Jones, C.: Gaps in the object-oriented paradigm. IEEE Computer, June 1994, pp. 90-91.
9. Neumann, K.: Concept and prototyp implementation of tools for the evaluation of software quality in object-oriented development (german). Study, TU Dresden / TU Magdeburg, January 1991.
10. Sharble, R.C.; Cohen, S.S.: The Object-Oriented Brewery: A Comparison of Two Object-Oriented Development Methods. Software Engineering Notes, 18(1993)2, pp. 60-73.
11. Stiebellehner, J. et. al.: Coupling and Cohesion Metrics for Object-Oriented Software. (german) in Dumke/Zuse: Theorie und Praxis der Softwaremessung, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp. 111-135.
12. Taylor, D.A.: A quality-first program for object technology. Object magazine, 2(1992), pp. 17-18.
13. Tegarden, D.P.; Sheetz, S.D.; Monarchi, D.E.: A Software Complexity Model of Object-Oriented Systems. University of Denver, 1992.

OBJECT-ORIENTED PROGRAMMING

1. Barnes, G.M.; Swim, B.R.: Inheritance software metrics. *Journal of object-oriented Programming*, Nov/Dec. 1993, pp. 27-34.
2. Berard, E.V.: *Essays on Object-Oriented Software Engineering*. Vol. I, Prentice Hall Inc., 1993.
3. Biemann, J.M.: *Deriving Measures of Software Reuse in Object Oriented Systems*. Technical Report CS-91-112, Colorado State University, July 1991.
4. Bradley, L.: *Evaluating Complex Properties of Object-Oriented Design and Code*. *Proceedings of the International Software Quality Conference*, Dayton, Ohio, October 7-9, 1991, pp. 32-36.
5. Buth, A.: *Software metrics for object-oriented programming languages (german)*. Study GMD 545, Birlinghoven, June 1991.
6. Churcher, N.I.; Shepperd, M.J.: *Towards a Conceptual Framework for Object-Oriented Software Metrics*. *Software Engineering Notes*, 20(1995)2, pp. 69-75.
7. Dorman, M.: *Unit TEsting in C++ Objects*. *First European International Conference on Software Testing, Analysis & Review (EuroStar)*, London, October 25-28, 1993, pp. 71-101.
8. Dumke, R.; Winkler, A.: *The Nonclassical Use of Software Metrics*. *Proceedings of the Second International Conference on Software Quality*, Research Triangle Park, NC, October 5-7, pp. 284-291.
9. Fetcke, T.: *Software Metrics in Object-Oriented Programming*. Master Thesis, GMD Bonn, 1995.
10. Karunanithi, S.; Bieman, J.M.: *Candidate Reuse Metrics For Object Oriented and Ada Software*. *Proceedings of the First International Software Metrics Symposium*, Baltimore, May 21-22, 1993, pp. 120-128.
11. Lake, A.; Cook, C.: *A Software Complexity Metric for C++*. *Proceedings of the Fourth Annual Workshop on Software Metrics*, Oregon, March 22-24, 1992.
12. Lake, A.; Cook, C.: *Use Factor Analysis to Develop OOP Software*. *Proceedings of the Annual Oregon Workshop on Software Metrics*, April 10-12, 1994, Silver Falls, Oregon.
13. LaLonde, W.; Pugh, J.: *Complexity in C++: A Smalltalk Perspective*. *JOOP*, March-April 1995, pp. 49-56.
14. LaLonde, W.; Pugh, J.: *Gathering metric information using metalevel facilities*. *JOOP*, March/April 1994, pp. 33-37.
15. Leijter, M.; Meyers, S.; Reiss, S.P.: *Support for Maintaining Object-Oriented Programs*. *IEEE Transaction on Software Engineering*, 18(1992)12, pp. 1045-1052.
16. Li, Wei; Henry, S.: *Maintenance Metrics for the Object Oriented Paradigm*. *Proceedings of the First International Software Metrics Symposium*, Baltimore, May 21-22, 1993, pp. 52-60.
17. Lorenz, M.: *Object-Oriented Software-Development*. Prentice Hall Inc., 1993.

18. Morschel, I.: Applying Object-Oriented Metrics to Enhance Software Quality. in: Dumke/Zuse: Theorie und Praxis der Softwaremessung, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp. 97-110.
19. Neumann, K. : Implementation of software metrics in Smalltalk (german). Master's Thesis, TU Magdeburg, 1992.
20. Overbeck, J.: Testing Object-Oriented Software - State of the Art and Research Directions. First European International Conference on Software Testing, Analysis & Review (EuroStar), London, October 25-28, 1993, pp. 245-270.
21. Ponder, C.; Bush, B.: Polymorphism Considered Harmful. SIGPLAN Notices, 27(1992)6, pp. 76-79.
22. Rajaraman, C.; Lyu, M.R.: Reliability and Maintainability Related Software Coupling Metrics in C++ Programs. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, 1992, pp. 303-311.
23. Rocacher, D.: Metrics Definition for Smalltalk. ESPRIT Project 1257 MUSE Q/R, CRIL, Puteaux, 1988.
24. Rocacher, D.: Smalltalk Measure Analysis Manual. ESPRIT Project 1257 MUSE WP9A, CRIL, Rennes, 1989.
25. Sakkinen, M.: The Darker Side of C++ Revisited. Structured Programming, 13(1992)4, pp. 155-177.
26. Wilde, N.; Huitt, R.: Maintenance Support for Object-Oriented Programs. IEEE Transactions on Software Engineering, 18(1992)12, pp. 1038-1044.

OBJECT-ORIENTED SOFTWARE ENGINEERING BIBLIOGRAPHY

1. Abbott, "Report on Teaching Ada," Technical Report SAI-81-313-WA, Science Applications, Inc., McClean, Virginia, 1980.
2. Abbott, "Program Design by Informal English Descriptions," Communications of the ACM, Vol. 26, No. 11, November 1983, pp. 882 - 894.
3. Abbott, An Integrated Approach to Software Development, John Wiley & Sons, New York, New York, 1986.
4. Abelson, G.J. Sussman, and J. Sussman, Structure and Interpretation of Computer Programs, MIT Press, Cambridge, Massachusetts, 1985.
5. Abiteboul and V. Vianu, "Procedural and Declarative Database Update Languages," Proceedings of ACM PODS 1988, pp. 240 - 250.
6. Ackroyd and D. Daum, "Graphical Notation for Object-Oriented Design and Programming," Journal of Object-Oriented Programming, Vol. 3, No. 5, January 1991, pp. 18 - 28.
7. Association for Computing Machinery, special issue of SIGPLAN Notices on the Object-Oriented Programming Workshop, Vol. 21, No. 10, October 1986.

8. Association for Computing Machinery, OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986.
9. Association for Computing Machinery, OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987.
10. Association for Computing Machinery, Special Issue on Object-Oriented Systems, ACM Transactions on Office Information Systems, Vol. 5, No. 1, January 1987.
11. Association for Computing Machinery, OOPSLA '87 Addendum to the Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 5, May 1988.
12. Association for Computing Machinery, OOPSLA '88 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988.
13. Association for Computing Machinery, special issue of SIGPLAN Notices: Proceedings of the ACM SIGPLAN Workshop on Object-Based Concurrent Programming, Vol. 24, No. 4, April 1989.
14. Association for Computing Machinery, OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989.
15. Association for Computing Machinery, OOPSLA/ECOOP '90 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990.
16. Association for Computing Machinery, OOPSLA '91 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 26, No. 11, November 1991.
17. Adams and D. Lenkov, "Object-Oriented COBOL: The Next Generation," Hotline on Object-Oriented Technology, Vol. 2, No. 2, pp. 12 - 15.
18. Adams and A.K. Nabi, "Neural Agents -- A Frame of Mind," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 139 - 150.
19. Adelson and E. Soloway, "The Role of Domain Experience in Software Design," IEEE Transactions on Software Engineering, Vol. SE-11, No. 11, November 1985, pp. 1351 - 1360.
20. Adrion, "Multiple Inheritance in CAD Data Models," Proceedings of the Third International Workshop on Software Specification and Design, August 1985, IEEE Catalog No. 85 CH2138-6, IEEE Computer Society Press, Washington, D.C., pp. 1 - 5.
21. Agerwala, "Putting Petri Nets to Work," IEEE Computer, Vol. 12, No. 12, December 1979, pp. 85 - 94.
22. Agha and C. Hewitt, "Actors: A Conceptual Foundation for Concurrent Object-Oriented Programming," in Research Directions in Object-Oriented Programming, Edited by B. Shriver and P. Wegner, MIT Press, Cambridge, Massachusetts, 1987. pp. 47 - 74.
23. Agha, "An Overview of Actor Languages," SIGPLAN Notices, Vol. 18, No. 6, June 1983, pp. 58 - 67.

24. Agha, ACTORS, A Model of Concurrent Computation in Distributed Systems, MIT Press, Cambridge, Massachusetts, 1986.
25. Agha, "Concurrent Object-Oriented Programming," Communications of the ACM, Vol. 33, No. 9, September 1990, pp. 125 - 141.
26. Agha, P. Wegner, and A. Yonezawa, Research Directions in Concurrent Object-Oriented Programming, The MIT Press, Cambridge, Massachusetts, 1993.
27. Ahlsen, A. Björnerstedt, and C. Hultén, "OPAL: An Object-Based System for Application Development," IEEE Database Engineering, Vol. 8, No. 4, pp. 31-40.
28. Ahmed, A. Wong, D. Sriam, and R. Logcher, A Comparison of Object-Oriented Database Management Systems for Engineering Applications, Research Report No. R91-12, Order Number IESL90-03, 91-03, Massachusetts Institute of Technology, Department of Civil Engineering, Cambridge, Massachusetts, May 1991.
29. Aho, R. Sethi, and J.D. Ullman, Compilers: Principles, Techniques, and Tools, Addison-Wesley Publishing Company, Reading, Massachusetts, 1986.
30. Ait-Kaci and R. Nasr, "Login: A Logic Programming Language With Built-In Inheritance," The Journal of Logic Programming, Vol. 3, No. 3, October 1986, pp. 185 - 215.
31. Akama, "Inheritance Hierarchy Mechanism in Prolog," in Logic Programming '86 (Lecture Notes in Computer Science, No. 264, Tokyo, Japan, October 1986), Springer-Verlag, New York, New York, 1986.
32. Aksit, J.W. Dijkstra, and A. Tripathi, "Atomic Delegation: Object-Oriented Transactions," IEEE Software, Vol. 8, No. 2, March 1991, pp. 84 - 92.
33. Alabiso, "Transformation of Data Flow Analysis Models to Object -Oriented Design," OOPSLA '88 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 335 - 353.
34. Alagic, Object-Oriented Database Programming, Springer-Verlag, New York, New York, 1989.
35. Alexandridis, "Adaptable Software and Hardware : Problems and Solutions," Computer, Vol. 19, No. 2, February 1986, pp. 29 - 39.
36. Almes and C.L. Holman, "Edmas: An Object-Oriented, Locally Distributed Mail System," IEEE Transactions on Software Engineering, Vol. SE-13, No. 9, September 1987, pp. 1001 - 1009.
37. Ambriola and L. Bendix, "Object-Oriented Configuration Control," Proceedings of the 2nd International Workshop on Software Configuration Management, special issue of Software Engineering Notes, Vol. 17, No. 7, November 1989, pp. 133 - 136.
38. Ambrose and K.L. Rogers, Lessons Learned: Object-Oriented Methodologies, Ada, The Data Management System Testbed, and Prototyping, September 1989, MITRE Technical Report MTR-88D00083, MITRE Corporation, McLean, Virginia, 1989.

39. America, "POOL-T: A Parallel Object-Oriented Language," in Object-Oriented Concurrent Programming, A. Yonezawa and M. Tokoro, Editors, MIT Press, Cambridge, Massachusetts, 1987, pp. 199 - 220.
40. America, "Inheritance and Subtyping In a Parallel Object-Oriented Language," ECOOP '87: Proceedings of the European Conference on Object-Oriented Programming, Lecture Notes on Computer Science, Volume 276, Springer Verlag, New York, New York, 1987. pp. 234 - 242.
41. America, Editor, ECOOP '91: European Conference on Object-Oriented Programming, Geneva, Switzerland, July 1991, Springer-Verlag, New York, New York, 1991.
42. Ammirati and M. Gerhardt, "Using Object-Oriented Thinking to Teach Ada," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 277 - 300.
43. Anderson, J. McDonald, L. Holland, and E. Scranage, "Automated Object-Oriented Requirements Analysis and Design," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 265 - 272.
44. Anderson, "Achieving Interoperability: Myth and Reality," Infosystems, Vol. 34, No. 7, July 1987, page 56.
45. Anderson, "Technology Insertion: Establishing an Object-Oriented Life-Cycle Methodology," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 163 - 171.
46. Antebi, "Issues In Teaching C++," Journal of Object-Oriented Programming, Vol. 3, No. 4, November/December 1990, pp. 11 - 14, 16 - 21.
47. Arango, "Domain Analysis: From Art to Engineering Discipline," Proceedings of the Fifth International Workshop On Software Specification and Design, May 19-20, 1989, Pittsburgh, Pennsylvania, IEEE Computer Society Press, Washington, D.C., May 1989, pp. 152 - 159.
48. Aranow, "Object Technology Means Object-Oriented Thinking," Software, Vol. 12, No. 3, March 1992, pp. 41 - 44, 46 - 48.
49. Arden, "Automatic Programming Systems for Ada: Theory and Practice of Object-Oriented Methods of Program Specification from Requirements," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 69 - 71.
50. Arnol, "An Ada Application: A Remote Maintenance Monitoring System for Radar Stations," Proceedings of the Third German Ada Users Congress, January 1988, Gesellschaft für Software Engineering, Munich, West Germany, pp. 11-1 - 11-5.
51. Air Training Command, Object-Oriented Design (Student Handout), USAF Technical Training School, Keesler AFB, Mississippi, May 1989.
52. Atkinson and O.P. Bunerman, "Types and Persistence in Database Programming Languages," ACM Computing Surveys, Vol. 19, No. 2, June 1987.
53. Atkinson, P.J. Bailey, W.P. Cockshott, K.J. Chisholm, and R. Morrison, "An Approach to Persistent Programming," Computer Journal, Vol. 26, No. 4, 1983, pp. 360 - 365.

54. Atkinson, W.P. Cockshott, and R. Marshall, "Algorithms for a Persistent Heap," *Software Practice and Experience*, Vol. 13, No. 3, March 1983, pp. 259 - 271.
55. Atkinson, F. Bancilhon, D. DeWitt, K. Dittrich, D. Maier, and S. Zdonik, "The Object-Oriented Database System Manifesto," (Invited Paper), *Proceedings of the First International Conference on Deductive and Object-Oriented Databases*, Kyoto, Japan, December 4-6, 1989, pp. 40 - 57.
56. Atkinson, *Object-Oriented Reuse, Concurrency and Distribution*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1991.
57. Atre, "The Scoop on OOPS," *ComputerWorld*, Vol. XXIV, No. 38, September 17, 1990, pp. 115 - 116.
58. Auer, "Which Object-Oriented Language Should We Choose?," *Hotline on Object-Oriented Technology*, Vol. 1, No. 1, pp. 1, 3 - 6.
59. Babcock, "Object is DBMS Focus," *ComputerWorld*, Vol. XXI, No. 40, October 5, 1987, page 25.
60. Babcock, "Object-Oriented Systems Emerge," *ComputerWorld*, Vol. XXII, No. 8, February 22, 1988, page 25.
61. Bach, "Is Ada Really an Object-Oriented Programming Language," *Proceedings of Ada Expo 1988*, Galaxy Productions, Frederick, Maryland, 1988, 7 pages.
62. Bailey, "Designing With Objects," *Computer Language*, Vol. 6, No. 1, January 1989, pp. 34 - 43.
63. Bailin, "An Object-Oriented Specification Method for Ada," *Proceedings of the Fifth Washington Ada Symposium*, June 27 - 30, 1988, Association for Computing Machinery, New York, New York, 1988, pp. 139 - 150.
64. Bailin, "Remarks on Object-Oriented Requirements Specification," *Methodologies and Tools for Real-Time Systems Conference Proceedings*, National Institute for Software Quality and Productivity, Washington, D.C., November 1988, pp. H1 - H11.
65. Bailin, "An Object-Oriented Requirements Specification Method," *Communications of the ACM*, Vol. 32, No. 5, May 1989, pp. 608 - 623.
66. Balin, "Object-Oriented Requirements Analysis," *The Encyclopedia of Software Engineering*, Volume 2, J.J. Marciniak, Editor, John Wiley and Sons, New York, New York, 1994, pp. 740 - 756.
67. Baker, "Compartmentalized, Object-Oriented Ada Programming," *Proceedings of the Seventh Washington Ada Symposium*, June 25-28, 1990, pp. 183 - 187.
68. Baker, "Object-Oriented Programming in Ada83 -- Genericity Rehabilitated," *Ada Letters*, Vol. XI, No. 9, November/December 1991, pp. 116 - 127.
69. Banerjee, H. Chou, J.F. Garza, W. Kim, D. Woelk, N. Ballou and H. Kim, "Data Model Issues for Object-Oriented Applications," *ACM Transactions on Office Information Systems*, Vol. 5, No. 1, January 1987, pp. 3 - 26.

70. Banerjee, W.K.H-J. Kim, and H.F. Korth, "Data Model Issues for Object-Oriented Applications," ACM Transactions on Office Information Systems, Vol. 5, No. 1, pp. 3 - 26.
71. Banerjee, W.K.H-J. Kim, and H.F. Korth, "Semantics and Implementation of Schema Evolution in Object-Oriented Databases," ACM SIGMOD Notices, Vol. 16, No. 3, December 1987, pp. 311 - 322.
72. Barr and E.A. Feigenbaum, Editors, The Handbook of Artificial Intelligence, Volume 1, HeurisTech Press, Stanford, California, 1981.
73. Barrett and J.D. Couch, Compiler Construction: Theory and Practice, Science Research Associates, Chicago, Illinois, 1979.
74. Barstow, "Domain-Specific Automatic Programming," IEEE Transactions on Software Engineering, Vol. SE-11, No. 11, November 1985, pp. 1321 - 1336.
75. Barth, "An Object-Oriented Approach to Graphical Interfaces," ACM Transactions on Graphics, Vol. 5, No. 2, April 1986, pp. 142 - 172.
76. Baskette, "Life-Cycle Analysis of an Ada Project," IEEE Software, Vol. 4, No. 1, January 1987, pp. 40 - 47.
77. Bauer and H. Wossner, Algorithmic Language and Program Development, Springer-Verlag, New York, New York, 1982.
78. Beaudet and M.A. Jenkins, "Simulating the Object-Oriented Paradigm to Nial," SIGPLAN Notices, Vol. 23, No. 6, June 1988, pp. 49 - 58.
79. Beaudouin-Lafon, Object-Oriented Languages: Basic Principles and Programming Techniques, Chapman Hall, London, United Kingdom, 1994.
80. Beck and W. Cunningham, "A Laboratory for Teaching Object Oriented Thinking," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 1 - 6.
81. Beck, "Think Like an Object," UNIX Review, Vol. 9, No. 10, October 1991, pp. 39 - 43.
82. Beech and B. Mahbod, "Generalized Version Control in an Object-Oriented Database," Proceedings of the 1988 IEEE Fourth International Conference on Data Engineering, IEEE Computer Society Press, Washington, D.C., February 1988, pp. 14 - 22.
83. Belz, "Applying the Spiral Model: Observations on Developing System Software in Ada," Proceedings of the 1986 Annual Conference on Ada Technology, Atlanta, Georgia, 1986, pp. 57 - 66.
84. Ben-Ari, Principles of Concurrent Programming, Prentice Hall, Englewood Cliffs, New Jersey, 1982.
85. Benington, "Production of Large Computer Programs," Proceedings of the ONR Symposium on Advanced Program Methods for Digital Computers, June 1956, pp. 15 - 27. Also available in Annals of the History of Computing, October 1983, pp. 350 - 361.
86. Berard, An Object-Oriented Design Handbook for Ada Software, EVB Software Engineering, Inc., Frederick, Maryland, 1985.

87. Berard, "Object-Oriented Development," Methodologies and Tools for Real-Time Systems Conference Proceedings, National Institute for Software Quality and Productivity, Washington, D.C., March 1987, pp. E1 - E27.
88. Berard, "Life-Cycle Approaches," Hotline On Object-Oriented Technology, Vol. 1, No. 6, April 1990, pp. 1, 3 - 4.
89. Berard, "Understanding the Recursive/Parallel Life-Cycle," Hotline On Object-Oriented Technology, Vol. 1, No. 7, May 1990, pp. 10 - 13.
90. Berard, "Object-Oriented Requirements Analysis," Hotline On Object-Oriented Technology, Vol. 1, No. 8, June 1990, pp. 9 - 11.
91. Berard, "Readings In Object-Oriented Technology," OOPS Newsletter, Issue 10, British Computer Society Specialist Group on Object-Oriented Programming and Systems, 1990, pp. 6 - 8.
92. Berard, "Motivation for an Object-Oriented Approach to Software Engineering," OOPS Newsletter, Issue 11, British Computer Society Specialist Group on Object-Oriented Programming and Systems, 1990, pp. 10 - 20.
93. Berard, Essays on Object-Oriented Software Engineering, Volume 1, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
94. Berard, "Object-Oriented Development," Methodologies and Tools for Real-Time Systems Conference Proceedings, National Institute for Software Quality and Productivity, Washington, D.C., March 1987, pp. E1 - E27.
95. Berard, "Life-Cycle Approaches," Hotline On Object-Oriented Technology, Vol. 1, No. 6, April 1990, pp. 1, 3 - 4.
96. Berard, "Understanding the Recursive/Parallel Life-Cycle," Hotline On Object-Oriented Technology, Vol. 1, No. 7, May 1990, pp. 10 - 13.
97. Berard, "Object-Oriented Requirements Analysis," Hotline On Object-Oriented Technology, Vol. 1, No. 8, June 1990, pp. 9 - 11.
98. Berard, "Readings In Object-Oriented Technology," OOPS Newsletter, Issue 10, British Computer Society Specialist Group on Object-Oriented Programming and Systems, 1990, pp. 6 - 8.
99. Berard, "Motivation for an Object-Oriented Approach to Software Engineering," OOPS Newsletter, Issue 11, British Computer Society Specialist Group on Object-Oriented Programming and Systems, 1990, pp. 10 - 20.
100. Berard, Essays on Object-Oriented Software Engineering, Volume 1, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
101. Berzins, M. Gray, and D. Naumann, "Abstraction-Based Software Development," Communications of the ACM, Vol. 29, No. 5, May 1986, pp. 402 - 415.

102. Bewtra, S.C. Balin, and J.M. Moore, "An Ada Design and Implementation Toolset Based on Object-Oriented and Functional Programming Paradigms," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 213 - 226.
103. Bézivin and B. Meyer, Editors, Technology of Object-Oriented Languages and Systems: Tools 4, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
104. Bézivin, J.-M. Hullot, P. Cointe, and H. Lieberman, ECOOP '87: Proceedings of the European Conference on Object-Oriented Programming, Lecture Notes on Computer Science, Volume 276, Springer Verlag, New York, New York, 1987.
105. Bhaskar, "How Object-Oriented Is Your System," SIGPLAN Notices, Vol. 18, No. 10, October 1983, pp. 8 - 11.
106. Biddle, "Methodology in Selecting Tools for a Real-time Embedded Ada Program," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 63 - 66.
107. Birchenough and J.R. Cameron, "JSD and Object-Oriented Design," JSP & JSD: The Jackson Approach to Software Development, IEEE Computer Society Press, Washington, D.C., 1989.
108. Birkhead, "The Object Objective: Servio Logic's GemStone for VMS Manages Database With Objects," DEC Professional, August 1989, page 30.
109. Birrell and M.A. Ould, A Practical Handbook for Software Development, Cambridge University Press, Cambridge, United Kingdom, 1985.
110. Birtwistle, O. Dahl, B. Myhrtag and K. Nygaard, Simula Begin, Auerbach Press, Philadelphia, 1973.
111. Bishop, "The Effect of Data Abstraction on Loop Programming Techniques," IEEE Transactions on Software Engineering, Vol. 16, No. 4, April 1990, pp. 389 - 402.
112. Bitman, "Functional Lists: Object-Oriented Design Classes for MIS Applications," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 101 - 122.
113. Bjørner, T. Denvir, E. Meiling, and J.S. Pederson, The RAISE Project: Fundamental Issues and Requirements, Document Number RAISE/DDC/EM/1/V6, Dansk Datamatik Center, Denmark, December, 1985.
114. Black, N. Hutchinson, E. Jul, and H. Levy, "Object Structure in the Emerald System," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 78 - 86.
115. Blaha, W.J. Premerlani, and J.E. Rumbaugh, "Relational Database Design Using an Object-Oriented Approach," Communications of the ACM, Vol. 31, No. 4, April 1988, pp. 414 - 427.
116. Blair, J.J. Gallagher, and J. Malik, "Genericity vs. Inheritance vs. Delegation vs. Conformance vs. ...," Journal of Object-Oriented Programming, Vol. 2, No. 3, September-October 1989, pp. 11 - 17.

117. Blair, J. Gallagher, D. Hutchison, and D. Shepherd, *Object-Oriented Languages, Systems and Applications*, Halsted Press, New York, New York, 1991.
118. Blake and P. Wisskirchen, Editors, *Advances In Object-Oriented Graphics I*, Springer-Verlag, New, York, New York, 1991.
119. Blank and M.J. Krijger, Editors, *Software Engineering: Methods and Techniques*, John Wiley & Sons, New York, New York, 1983.
120. Bloor, "Object Orientation Has the Edge Over Leading Languages," *DEC User*, October 1989, pp. 55 - 56.
121. Blumberg, J. Kantor, M. McNickle, and A. Reedy, "NASA Software Support Environment: Configuring an Environment for Ada Design," in *Ada In Industry: Proceedings of the Ada-Europe International Conference Munich 7-9 June, 1988*, Cambridge University Press, Cambridge, United Kingdom, 1988, pp. 3 - 16.
122. Boar, *Applications Prototyping*, John Wiley & Sons, New York, New York, 1984.
123. Bobrow, K. Kahn, G. Kiczales, L. Masinter, M. Stefik, and F. Zdybel, "CommonLoops: Merging Lisp and Object-Oriented Programming," *OOPSLA '86 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 21, No. 11, November 1986, pp. 17 - 29.
124. Bobrow, L.G. DeMichiel, R.P. Gabriel, S.E. Keene, G. Kiczales, and D.A. Moon, "Common LISP Object System Specification X3J13 Document 88-002R," *SIGPLAN Notices*, Vol. 23, Special Issue, September 1988.
125. Bobrow, *The LOOPS Manual*, Rank Xerox, Inc., Palo Alto, California, 1983.
126. Bochenski, "On Object-Oriented Programming, Databases," *Software*, Vol. 8, No. 11, September 1988, page 42.
127. Boehm and F.C. Belz, "Applying Programming to the Spiral Model," *Proceedings of the 4th International Software Process Workshop*, May 1988, Special Issue of the *ACM SIGSoft Software Engineering Notes*, Vol. 14, No. 4, June 1989, pp. 46 - 56.
128. Boehm, T.L. Gray, and T. Seewaldt, "Prototyping Versus Specifying: A Multiproject Experiment," *IEEE Transactions on Software Engineering*, Vol. SE-10, No. 3, May 1984, pp. 290 - 302.
129. Boehm, "Software Engineering," *IEEE Transactions on Computers*, Vol. 25, No. 12, December 1976, pp. 1226 - 1241.
130. Boehm, *Software Engineering Economics*, Prentice Hall, Englewood Cliffs, New Jersey, 1981.
131. Boehm, "A Spiral Model of Development and Enhancement," *Software Engineering Notes*, Vol. 11, No. 4, August, 1986, pp. 14 - 24.
132. Boehm-Davis and L.S. Ross, "Approaches to Structuring the Software Development Process," *General Electric Company Report Number GEC/DIS/TR-84-B1V-1*, October 1984.

133. Booch and M. Vilot, "The Design of the C++ Booch Components," OOPSLA/ECOOP '90 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990, pp. 1 - 11.
134. Booch, "Describing Software Design in Ada," SIGPLAN Notices, Vol. 16, No. 9, September 1981, pp. 42 - 47.
135. Booch, "Object Oriented Design," Ada Letters, Vol. I, No. 3, March- April 1982, pp. 64 - 76.
136. Booch, "Solve Process-Control Problems With Ada's Special Capabilities," EDN, June 23, 1982, pp. 143 - 152.
137. Booch, Software Engineering with Ada, Benjamin/Cummings, Menlo Park, California, 1983.
138. Booch, "Object Oriented Design," IEEE Tutorial on Software Design Techniques, Fourth Edition, P. Freeman and A.I. Wasserman, Editors, IEEE Computer Society Press, IEEE Catalog No. EHO205-5, IEEE-CS Order No. 514, pp. 420 - 436.
139. Booch, "Object Oriented Development," IEEE Transactions on Software Engineering, Vol. SE-12, No. 2, February 1986, pp. 211 - 221.
140. Booch, Software Engineering with Ada, Second Edition, Benjamin/Cummings, Menlo Park, California, 1986.
141. Booch, Software Components With Ada, Benjamin/Cummings, Menlo Park, California, 1987.
142. Booch, "On the Concepts of Object-Oriented Design," in Modern Software Engineering: Foundations and Current Perspectives, P.A. Ng and R.T. Yeh, Editors, Van Nostrand Reinhold, New York, New York, 1990, pp. 165 - 204.
143. Booch, Object-Oriented Design With Applications, Benjamin/Cummings, Menlo Park, California, 1991.
144. Booch, Object-Oriented Analysis and Design With Applications, Second Edition, Benjamin/Cummings, Menlo Park, California, 1991.
145. Booch, "Coming of Age In an Object-Oriented World," IEEE Software, Vol. 11, No. 6, November 1994, pp. 33 - 41.
146. Booch and M. Vilot, "Object-Oriented Design: Inheritance Relationships," The C++ Report, Vol. 2, No. 9, October 1990, pp. 8 - 11.
147. Borgida, S. Greenspan, and J. Mylopoulos, "Knowledge Requirements as the Basis for Requirements," IEEE Computer, Vol. 18, No. 4, April 1985, pp. 82 - 91.
148. Borgida, "Language Features for Flexible Handling of Exceptions in Information Systems," ACM Transactions on Database Systems, Vol. 10, No. 4, December 1985, pp. 565 - 603.
149. Borgida, "Exceptions in Object-Oriented Languages," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 107 - 119.

150. Borning and D.H.H. Ingalls, "Multiple Inheritance in Smalltalk-80," Proceedings of the National Conference on Artificial Intelligence, August 1982, pp. 234 - 237.
151. Borning and D.H.H. Ingalls, "A Type Declaration and Inference System for Smalltalk," 9th Annual ACM Symposium on Principles of Programming Languages, Albuquerque, New Mexico, 1982, pp. 133 - 141.
152. Borning, ThingLab -- A Constraint-Oriented Simulation Laboratory (Ph.D. Dissertation), Stanford University, July 1979, published as Technical Report SSL-79-3, Xerox Palo Alto Research Center, Palo Alto, California.
153. Borning, "The Programming Language Aspects of ThingLab," ACM Transactions on Programming Languages, Vol. 3, No. 4, October 1981, pp. 353 - 387.
154. Borning, "Class Versus Prototypes in Object-Oriented Languages," Proceedings of the 1986 Fall Joint Computer Conference, IEEE Catalog Number 86CH2345-7, IEEE Computer Society Press, Washington, D.C., 1986, pp 36 - 40.
155. Boyd, "Object-, Process-, and Behavior-Oriented Methods for Ada," Methodologies and Tools for Real-Time Systems Conference Proceedings, March 1987, National Institute for Software Quality and Productivity, Washington, D.C., pp. M1 - M13.
156. Boyd, "Object-Oriented Design and PAMELA: A Comparison of Two Design Methods for Ada," Ada Letters, Vol. 7, No. 4, July-August 1987, pp. 68 - 78.
157. Bozman, "Versant Is Dreaming Big," ComputerWorld, Vol. XXIV, No. 26, June 25, 1990, page 98.
158. Bozman, "Firms Move Into the OOP Lane," ComputerWorld, Vol. XXIV, No. 30, July 23, 1990, pp. 23 and 31.
159. Bozman, "They're Still Saying OOPs," ComputerWorld, Vol. XXIV, No. 34, August 20, 1990, pp. 29 and 34.
160. Bracha and W. Cook, "Mixin-Based Inheritance," OOPSLA/ECOOP '90 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990, pp. 303 - 311.
161. Britton, J. Doake, and R. Mitchell, "Taming the Abstract Data Type," Software Engineering Notes, Vol. 15, No. 1, January 1990, pp. 36 - 41.
162. Brookman, "SA/SD vs. OOD," Ada Letters, Vol. XI, No. 9, November/December 1991, pp. 96 - 99.
163. Brooks, Jr., The Mythical Man-Month, Addison-Wesley Publishing Company, Reading, Massachusetts, 1975.
164. Brooks, Jr., "No Silver Bullet: Essence and Accidents of Software Engineering," IEEE Computer, Vol. 20, No. 4, April 1987, pp. 10 - 19.
165. Brophy, W.W. Agresti, and V.R. Basili, "Lessons Learned in Use of Ada-Oriented Design Methods," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 231 - 236.

166. Brown and V. Dobbs, "A Method for Translating Functional Requirements for Object-Oriented Design," Proceedings of the Seventh Annual National Conference on Ada Technology, March 1989, pp. 589 - 599.
167. Brown and R.B. Quanrud, "The Generic Architecture Approach to Reusable Software," Proceedings of the Sixth National Conference on Ada Technology, March 14-18, 1988, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 390 - 394.
168. Brown, Object-Oriented Databases: Applications in Software Engineering, McGraw-Hill, New York, New York, 1991.
169. Bruno and A. Balsamo, "Petri Net-Based Object-Oriented Modeling of Distributed Applications," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 284 - 293.
170. Bruno and G. Marchetto, "A Methodology Based on High-Level Petri Nets for the Specification and Design of Control Systems," Proceedings of the Third International Workshop on Software Specification and Design, August 1985, IEEE Catalog No. 85 CH2138-6, IEEE Computer Society Press, Washington, D.C., pp. 30 - 34.
171. Budd, A Little Smalltalk, Addison-Wesley Publishing Company, Reading, Massachusetts, 1987.
172. Budd, An Introduction to Object-Oriented Programming, Addison-Wesley, Reading, Massachusetts, 1991.
173. Buhr and C.R. Zarnke, "Nesting In an Object-Oriented Language Is Not for the Birds," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 128 - 145.
174. Buhr, G. Ditchfield, and C.R. Zarnke, "Adding Concurrency to a Statically Type-Safe Object-Oriented Programming Language," Proceedings of the ACM SIGPLAN Workshop on Object-Based Concurrent Programming, San Diego, California, September 26-27, 1988, pp. 18 21.
175. Buhr, System Design With Ada, Prentice Hall, Englewood Cliffs, New Jersey, 1984.
176. Buhr, Practical Visual Techniques in System Design: With Applications to Ada, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
177. Bulman, "An Object-Based Development Model," Computer Language, Vol. 6, No. 8, August 1989, pp. 49 - 59.
178. Bulman, "Objects Don't Replace Design," Computer Language, Vol. 6, No. 8, August 1989, pp. 151 - 152.
179. Bulman, "Refining Candidate Objects," Computer Language, Vol. 8, No. 1, January 1991, pp. 30 - 39.
180. Buzzard and T.N. Mudge, "Object-Based Computing and the Ada Programming Language," IEEE Computer, Vol. 18, No. 3, March 1985, pp. 12 - 19.

181. Byrne and E. Wiatrowski, "Object-Oriented Design With Graphical Abstraction," Proceedings of the Third National Conference on Methodologies and Tools for Real-Time Systems, National Institute for Software Quality and Productivity, Washington, D.C., September 1986, pp. C-1 to C-19.
182. Caine and E.K. Gordon, "PDL -- A Tool for Software Design," Proceedings of the National Computer Conference, 1975, pp. 271 - 276.
183. Cameron, "Efficient High-Level Iteration With Accumulators," ACM Transactions on Programming Language Systems, Vol. 11, No. 2, April 1989, pp. 194 - 211.
184. Campos and G. Estrin, "SARA Aided Design of Software for Concurrent Systems," Proceedings of the National Computer Conference, AFIPS Press, Montvale, New Jersey, 1978, pp. 325 - 336.
185. Canning, W.R. Cook, W.L. Hill, J. Mitchell, and W. Olthoff, "F-Bounded Polymorphism for Object-Oriented Programming," Proceedings of the Conference on Functional Programming Languages and Computer Architecture, 1989, pp. 273 - 280.
186. Canning, W.R. Cook, W.L. Hill, and W.G. Olthoff, "Interfaces for Strongly-Typed Object-Oriented Programming," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, 457 - 467.
187. M Caplinger, "An Information System Based on Distributed Objects," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 126 - 137.
188. Cardelli and P. Wegner, "On Understanding Types, Data Abstraction, and Polymorphism," ACM Computing Surveys, Vol. 17, No. 4, December 1985, pp. 471 - 522.
189. Cardelli, "A Semantics of Multiple Inheritance," in Semantics of Data Types, Lecture Notes in Computer Science Volume 173, G. Kahn, D.B. McQueen, and G. Plotkin, Editors, Springer-Verlag, New York, New York, 1984, pp. 51 - 67.
190. Cardenas and D. McLeod, Editors, Research Foundations in Object-Oriented and Semantic Database Systems, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
191. Carey, D. Dewitt, and S. Vandenberg, "A Data Model and Query Language for EXODUS," Proceedings of ACM SIGMOD 1988, pp. 413 - 423.
192. Cargill, "Does C++ Really Need Multiple Inheritance," Proceedings of the C++ Conference, San Francisco, California, April 1990, USENIX Association, Berkeley, California, 1990, pp. 315 - 323.
193. Carlson, "Problems Encountered in Learning Object-Oriented Design Using Ada," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 209 - 212.
194. Carmichael, Object Development Methods, SIGS Books, New York, New York, 1994.
195. Caromel, "A General Model for Concurrent and Distributed Object-Oriented Programming," SIGPLAN Notices, Vol. 24, No. 4, April 1989, 102 - 104.

196. Caromel, "Service, Asynchrony, and Wait-By-Necessity," *Journal of Object-Oriented Programming*, Vol. 2, No. 4, November/December, 1989, pp. 12 - 22.
197. Caromel, "Concurrency: An Object-Oriented Approach," *Technology of Object-Oriented Languages and Systems 2 (TOOLS 2)*, 1990, pp. 183 - 197.
198. Carstensen, "A Real Example of Reusing Ada Software," *Proceedings of the Second National Conference on Software Reusability and Maintainability*, National Institute for Software Quality and Productivity, Washington, D.C., March 1987, pp. B-1 to B-19.
199. Casais, "An Object-Oriented System Implementing KNOs," *Proceedings of the Conference on Office Information Systems (COIS)*, Palo Alto, March 1988, pp. 284 - 290.
200. Cashman, "Object-Oriented Domain Analysis," *Software Engineering Notes*, Vol. 14, No. 6, October 1989, page 67.
201. Cattell, *Object Data Management: Object-Oriented and Extended Relational Database Systems*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1991.
202. Cattell and J. Skeen, "Object Operations Benchmark," *ACM Transactions on Database Systems*, Vol. 17, No. 1, March 1992, pp. 1 - 31.
203. Chambers and D. Ungar, "Customization: Optimizing Compiler Technology for Self, a Dynamically-Typed Object-Oriented Language," *SIGPLAN Notices*, Vol. 24, No. 7, July 1989, pp. 146 - 160.
204. Chambers, D. Ungar, and E. Lee, "An Efficient Implementation of Self, a Dynamically-Typed Object-Oriented Language Based on Prototypes," *OOPSLA '89 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 24, No. 10, October 1989, pp. 49 - 70.
205. Chan and B. Henderson-Sellers, "Corporate O-O Development Environment," *Software Engineering Notes*, Vol. 15, No. 1, January 1990, pp. 42 - 43.
206. Chan and W. Tsung-Juang, "What is the Object In Object-Oriented Programming," *Proceedings of the Seventh Annual National Conference on Ada Technology*, March 13-16, 1989, pp. 193 - 195.
207. Chan, "Lessons in Software Reusability in Large Complex Software Systems," *Proceedings of the Conference on Software Reusability and Portability*, National Institute for Software Quality and Productivity, Washington, D.C., September 16-17 1987, pp. B-1 to B-7.
208. Charniak and D. McDermott, *Introduction to Artificial Intelligence*, Addison-Wesley, Reading, Massachusetts, 1985.
209. Chedgey, S. Kerney, and H.-J. Kugler, "Using VDM in an Object-Oriented Development Method for Ada Software," *VDM '87 VDM -- A Formal Method At Work*, *Proceedings of the 1987 European Symposium*, Springer Verlag Lecture Notes On Computer Science, Number 252, pp. 63 - 76.
210. P.P-S Chen, "The Entity-Relationship Model -- Toward a Unified View of Data," *Transactions on Database Systems*, Vol. 1, No. 1, March 1976, pp. 9 - 36.

211. Chen, "An Object/Task Modeling Approach," Proceedings of the 1986 Fall Joint Computer Conference, IEEE Catalog Number 86CH2345-7, IEEE Computer Society Press, Washington, D.C., 1986, pp. 200 - 206.
212. Cherry, "Covering the Life Cycle with Ada: Ada All the Way," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 221 - 263.
213. Cherry, "System Construction With Object-Oriented Pictures," Software Engineering Notes, Vol. 16, No. 4, October 1991, pp. 42 - 51.
214. Chidamber and C.F. Kemerer, "Towards a Metrics Suite for Object-Oriented Design," OOPSLA '91 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 26, No. 11, November 1991, pp. 197 - 211.
215. Chignell, "Intelligent Databases and Object-Oriented Languages," Computer Language, Vol. 6, No. 10, October 1989, pp. 67 - 78.
216. Clark, "Designing Concurrent Objects," Ada Letters, Vol. VII, No. 6, Fall 1987, pp. 107 - 109.
217. Cluet, C. Delobel, C. Lecluse, and P. Richard, "Reloop, An Algebra Based Query Language for an Object-Oriented Database System," Proceedings of the First International Conference on Deductive and Object-Oriented Databases, Kyoto, Japan, December 4-6, 1989, pp. 294 - 313.
218. Coad and J. Nicola, Object-Oriented Programming, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
219. Coad and E. Yourdon, OOA -- Object-Oriented Analysis, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
220. Coad and E. Yourdon, OOA -- Object-Oriented Analysis, 2nd Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
221. Coad and E. Yourdon, Object-Oriented Design, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
222. Coad, "Object-Oriented Requirements Analysis (OORA): A Practitioner's Crib Sheet," Proceedings of Ada Expo 1988, Galaxy Productions, Frederick, Maryland, 1988, 9 pages.
223. Coad, "Object-Oriented Patterns," Communications of the ACM, Vol. 35, No. 9, September 1992, pp. 152 - 159.
224. Coad, D. North, and M. Mayfield, Object Models: Strategies, Patterns, and Applications, Prentice Hall, Englewood Cliffs, New Jersey, 1995.
225. Coggins, "Designing C++ Class Libraries," Proceedings of the C++ Conference, San Francisco, California, April 1990, USENIX Association, Berkeley, California, 1990, pp. 25 - 35.
226. Cohen, "Data Abstraction, Data Encapsulation, and Object-Oriented Programming," SIGPLAN Notices, Vol. 19, No. 1, January 1984, pp. 31 - 35.

227. Cointe, "Metaclasses are First Class: the ObjVLisp Model," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 156 - 167.
228. Cointe, "A Tutorial Introduction to Metaclass Architectures As Provided by Class Oriented Languages," International Conference on Fifth Generation Computer Systems (FGCS '88), Vol. 2, November-December 1988, Icot, Tokyo, Japan, pp. 592 - 608.
229. Colbert, "The Object-Oriented Software Development Method: A Practical Approach to Object-Oriented Development," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 400 - 415.
230. Coleman, F. Hayes, and S. Bear, "Introducing Objectcharts or How to Use Statecharts in Object-Oriented Design," IEEE Transactions on Software Engineering, Vol. 16, No. 1, January 1992, pp. 9 - 18.
231. Coleman, P. Arnold, S. Bodoff, C. Dollin, H. Gilchrist, F. Hayes, and P. Jeremaes, Object-Oriented Development: The Fusion Method, Prentice Hall, Englewood Cliffs, New Jersey, 1994.
232. Collard, "Object-Oriented Programming Techniques With Ada: An Example," Ada Letters, Vol. 9, No. 6, September/October 1989, pp. 119 - 126.
233. Collins, "What Is an Object-Oriented User Interface?", Proceedings of the Symposium on Object-Oriented Programming Emphasizing Practical Applications, Association for Computing Machinery Press, pp. 269 - 306.
234. Conery, "Logical Objects," in Proceedings of the 5th International Conference/Symposium on Logic Programming, Seattle, Washington, August 1988, MIT Press, Cambridge, Massachusetts, pp. 470 - 474.
235. Constantine, "Object-Oriented and Structured Methods: Towards Integration," American Programmer, Vol. 2, No. 7-8, August 1989, pp. 34 - 40.
236. Constantine, "Objects, Functions, and Program Extensibility," Computer Language, Vol. 7, No. 1, January 1990, pp. 34 - 56.
237. Constantine, "Larry Constantine on Structured Methods and Object Orientation," UNIX Review, Vol. 9, No. 2, February 1991, pp. 30 - 31.
238. Cook and J. Palsberg, "A Denotational Semantics of Inheritance and Its Correctness," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 433 - 443.
239. Cook, W. Hill, and P. Canning, "Inheritance Is Not Subtyping," Report STL-89-17 (Revision 1), Hewlett-Packard Laboratories, Palo Alto, California, 1989, 11 pages. Also in the Proceedings of the Seventeenth Symposium on Principles of Programming Languages, January 1990, pp. 125 - 135.
240. Cook, "Languages and Object-Oriented Programming," Software Engineering Journal, Vol. 1, No. 2, 1986, pp. 73 - 80.

- 241. Cook, "OOPSLA '87 Panel P2: Varieties of Inheritance," OOPSLA '87 Addendum to the Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 5, May 1988, pp. 35 - 40.
- 242. Cook, A Denotational Semantics of Inheritance (Ph.D. Thesis), Technical Report No. CS-89-33, Department of Computer Science, Brown University, Providence, Rhode Island, May 15, 1989.
- 243. Cook, Editor, ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989.
- 244. Coplien, Advanced C++: Programming Styles and Idioms, Addison-Wesley, Reading, Massachusetts, 1991.
- 245. Cortese, "Building Blocks," ComputerWorld, Vol. 23, No. 14, April 3, 1989, page 23.
- 246. Cortese, "Object-Oriented Standards Group Adopts HP's New Wave," ComputerWorld, Vol. 23, No. 17, April 24, 1989, page 12.
- 247. Courtois, "On Time and Space Decomposition of Complex Structures," Communications of the ACM, Vol. 28, No. 6, June 1985, pp. 590 - 603.
- 248. Cox and A.J. Navobilski, Object Oriented Programming: An Evolutionary Approach, Second Edition, Addison-Wesley, Reading, Massachusetts, 1991.
- 249. Cox and T. Pietrzykowski, "Using a Pictorial Representation to Combine Data Flow and Object-Orientation in a Language-Independent Programming Mechanism," Proceedings of the International Computer Science Conference, Hong Kong, December 1988, pp. 695 - 704.
- 250. Cox and T. Pietrzykowski, "User-Oriented Software: A New Methodology for Software Development," Computer Language, Vol. 6, No. 9, September 1989, pp. 79 - 92.
- 251. Cox, "The Message/Object Programming Model: A Small Change at a Deep Conceptual Level," in IEEE Proceedings of the Softfair: A Conference on Software Development Tools, Techniques, and Alternatives, July 25-28, 1983 pp. 51 - 60.
- 252. Cox, "Message/Object Programming: An Evolutionary Change In Programming Technology," IEEE Software, Vol. 1, No. 1, January 1984, pp. 50 - 61.
- 253. Cox, Object Oriented Programming: An Evolutionary Approach, Addison-Wesley, Reading, Massachusetts, 1986.
- 254. Cox, "There Is a Silver Bullet," Byte, Vol. 15, No. 10, October 1990, pp. 209 - 210, 212, 214, 216, 218.
- 255. Craske, "SNOOPS: An Object-Oriented Language Enhancement Supporting Dynamic Program Reconfiguration," ACM SIGPLAN Notices, Vol. 26, No. 10, October 1991, pp. 53 - 62.
- 256. Cunningham and K. Beck, "A Diagram for Object-Oriented Programs," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, pp. 361 - 367.

257. Curry and R.M. Ayers, "Experience with Traits in the Xerox Star Workstation," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 519 - 527.
258. Curry, "An Approach to Type Safety in a Traits System," Proceedings of the 1986 Fall Joint Computer Conference, IEEE Catalog Number 86CH2345-7, IEEE Computer Society Press, Washington, D.C., 1986, pp 25 - 30.
259. Curry, L. Baer, D. Lipkie and B. Lee, "TRAITS: an Approach for Multiple Inheritance Subclassing," Proceedings of the ACM SIGOA, SIGOA Newsletter, Vol. 3, No. 12, Philadelphia, June 1982.
260. Dahl and K. Nygaard, "SIMULA -- an ALGOL-Based Simulation Language," Communications of the ACM, Vol. 9, No. 9, September 1966, pp. 671 - 678.
261. Danforth and C. Tomlinson, "Type Theories and Object-Oriented Programming," ACM Computing Surveys, Vol. 20, No. 1, March 1988, pp. 29 - 72.
262. Davis and M. Irving, "Practical Experiences of Ada and Object-Oriented Design In Real-Time Distributed Systems," Ada: the Design Choice -- Proceedings of the Ada-Europe Conference, Madrid 13-15 June 1989, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 59 - 79.
263. Davis, "A Comparison of Techniques for the Specification of External System Behavior," Communications of the ACM, Vol. 31, No. 9, September 1988, pp. 1098 - 1115.
264. de Bondeli, "Real-Time Ada Systems: Development Methodology and Real-Time Performance," Ada Letters, Vol. VII, No. 6, Fall 1987, pp. 119 - 120.
265. de Champeaux and P. Faure, "A Comparative Study of Object-Oriented Analysis Methods," Journal of Object-Oriented Programming, Vol. 5, No. 1, March/April 1992, pp. 21 - 33.
266. de Champeaux, D. Lea, and P. Faure, Object-Oriented System Development, Addison-Wesley Publishing Company, Reading, Massachusetts, 1993.
267. Dearle, G.M. Shaw, and S.B. Zdonik, Implementing Persistent Object Bases, Principles and Practices: The Fourth International Workshop on Persistent Object Systems, Morgan Kaufman Publishers, Inc., San Mateo, California, 1991.
268. deBono, deBono's Thinking Course, Facts On File Publications, New York, New York, 1985.
269. Decouchant, S. Krakowiak, M. Meysembourg, M. Riveill, and X. de Pina, "A Synchronization Mechanism for Typed Objects In a Distributed System," SIGPLAN Notices, Vol. 24, No. 4, April 1989, pp. 105, 107.
270. Degano and E. Sandewall, Editors, Integrated Interactive Computing Systems, North Holland, New York, New York, 1983.
271. Delatte, M. Heitz, and J.F. Muller, HOOD Reference Manual 3.1, Prentice Hall, London, United Kingdom, 1993.
272. DeMarco, Structured Analysis and System Specification, Yourdon Press, New York, New York, 1979.

273. DeMichiel and R.P. Gabriel, "The Common Lisp Object System: An Overview," ECOOP '87: Proceedings of the European Conference on Object-Oriented Programming, Lecture Notes on Computer Science, Volume 276, Springer Verlag, New York, New York, 1987, pp. 151 - 170.
274. DeRemer and H.H. Kron, "Programming-in-the-Large Versus Programming-in-the-Small," IEEE Transactions on Software Engineering, Vol. SE-2, No. 2, June 1976, pp. 80 - 86, reprinted in IEEE Tutorial: Software Design Techniques, Third Edition, edited by P. Freeman and A.I. Wasserman, IEEE Catalog Number EHO161-0, IEEE Computer Society Press, Silver Spring, Maryland, 1983, pp. 237 - 243.
275. Detlefs, M.P. Herlihy, and J.M. Wing, "Inheritance of Synchronization and Recovery Properties on Avalon/C++," IEEE Computer, Vol. 21, No. 12, December 1988, pp. 57 - 69.
276. Dewan, "Object-Oriented Editor Generation" Journal of Object-Oriented Programming, Vol. 3, No. 2, July/August 1990, pp. 35 - 49.
277. Dhar and M. Jarke, "Dependency Directed Reasoning and Learning in Systems Maintenance Support," IEEE Transactions on Software Engineering, Vol. 14, No. 2, February 1988, pp. 211 - 227.
278. Di Maio, C. Cardigno, R. Bayan, C. Destombes, and C. Atkinson, "DRAGOON: An Ada-Based Object-Oriented Language," Ada: the Design Choice -- Proceedings of the Ada-Europe Conference, Madrid 13-15 June 1989, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 39 - 48.
279. Dietrich, Jr., L.R. Nackman, and F. Gracer, "Saving a Legacy With Objects," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 77 - 84.
280. Dijkstra, "Structure of the 'THE'-Multiprogramming System," Communications of the ACM, Vol. 11, No. 5, May 1968, pp. 341-346.
281. Dijkstra, "Structured Programming," originally appeared in a report on a conference sponsored by the NATO Science Committee, Rome, Italy, October 1969, reprinted in Classics in Software Engineering, Edited by Edward N. Yourdon, Yourdon Press, New York, New York, 1979, pp. 43 - 48.
282. Dittrich and U. Dayal, Editors, Proceedings of the 1986 International Workshop on Object-Oriented Database Systems, IEEE Catalog Number 86TH0161-0, IEEE Computer Society Press, Washington, D.C., 1986.
283. Dittrich, U. Dayal, and A.P. Buchmann, Editors, On Object-Oriented Database Systems, Springer-Verlag, New York, New York, 1991.
284. Dittrich, Editor, Advances in Object-Oriented Database Systems, Springer-Verlag, New York, New York, 1989.
285. Dlugosz, "Libraries With Class," Byte, Vol. 16, No. 2, February 1991, pp. 164 - 166, 168.
286. -E. Doberkat, "Tangraml -- A Program Description Language for Ada," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 390 - 403.

287. Dock, "OOD: Research Or Ready," Hotline On Object-Oriented Technology, Vol. 3, No. 9, July 1992, pp. 1, 7 - 9.
288. Military Standard Specification Practices (MIL-STD-490), Government Printing Office, Washington, D.C., 1968.
289. Department of Defense Requirements for High Order Computer Programming Languages: "Steelman", NTIS Order Number ADA059444, 1978.
290. Reference Manual for the Ada Programming Language, (ANSI/MIL-STD-1815A-1983), Government Printing Office, Washington, D.C., 1983.
291. Military Standard Software Development (DOD-STD-1679A (NAVY)), Government Printing Office, Washington, D.C., 1983.
292. Military Standard, Defense System Software Development (DOD-STD-2167), AMSC No. 3608, available from: Commander, Space and Naval Warfare Systems Command, Attention S&WS 811, Washington, D.C. 20363-5100, June, 1985.
293. British Department of Industry, Report of the Study of an Ada-Based System Development Methodology, Department of Industry (UK), 1981.
294. Donaldson, E.R. Comer, and A. Rudmik, "Ada Box Structures Starting With Objects," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 123 - 132.
295. Donaldson, "Dynamic Binding and Inheritance in an Object-Oriented Ada Design," Ada: the Design Choice -- Proceedings of the Ada-Europe Conference, Madrid 13-15 June 1989, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 16 - 25.
296. Dony, "An Object-Oriented Exception Handling System for an Object-Oriented Language," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 146 - 161.
297. Dony, "Exception Handling and Object-Oriented Programming: Towards a Synthesis," OOPSLA/ECOOP '90 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990, pp. 322 - 330.
298. Ducournau and M. Habib, "On Some Algorithms for Multiple Inheritance in Object-Oriented Programming," ECOOP '87: Proceedings of the European Conference on Object-Oriented Programming, Lecture Notes on Computer Science, Volume 276, Springer Verlag, New York, New York, 1987, pp. 243 - 252.
299. Duff and B. Howard, "Migration Patterns," Byte, Vol. 15, No. 10, October 1990, pp. 223 - 224, 226-228, 230, 232.
300. Duff, "Designing an Efficient Language," Byte, Vol. 11, No. 8, August 1986, pp. 211 - 224.
301. Duhl and C. Damon, "A Performance Comparison of Object and Relational Databases Using the Sun Benchmark," OOPSLA '88 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 153 - 163.

302. Duke, P. King, G. Rose, and G. Smith, "The Object-Z Specification Language," *Technology of Object-Oriented Languages and Systems: Tools 5*, Prentice Hall, Englewood Cliffs, New Jersey, 1991, pp. 465 - 483.
303. Duntemann and C. Marinacci, "New Objects for Old Structures," *Byte*, Vol. 15, No. 4, April 1990, pp. 261 - 266.
304. Eckart, "Iteration and Abstract Data Types," *SIGPLAN Notices*, Vol. 22, No. 4, April 1987, pp. 103 - 110.
305. Eckel, *Using C++*, Osborne McGraw-Hill, Berkeley, California, 1989.
306. Edelson, "How Objective Mechanisms Facilitate the Development of Large Software Systems in Three Programming Languages," *SIGPLAN Notices*, Vol. 22, No. 9, September 1987, pp. 54 - 63.
307. Egbert and W.J. Kubitz, "Application Graphics Modeling Support Through Object Orientation," *IEEE Computer*, Vol. 25, No. 10, October 1992, pp. 84 - 91.
308. Ege, *Programming in an Object-Oriented Environment*, Academic Press, Inc., San Diego, California, 1992.
309. Eliëns, *Principles of Object-Oriented Software Development*, Addison-Wesley, Reading, Massachusetts, 1995.
310. Eliot and F. Holliday, "Go Directly to Ada," *Computer Language*, Vol. 5, No. 5, May 1988, pp. 55 - 62.
311. Ellis and B. Stroustrup, *The Annotated C++ Reference Manual*, Addison-Wesley, Reading, Massachusetts, 1990.
312. Embley and S.N. Woodfield, "A Knowledge Structure for Reusing Abstract Data Types in Ada Software Production," *Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium*, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 27 - 34.
313. Embley, B.D. Kurtz, and S.N. Woodfield, *Object-Oriented Systems Analysis: A Model-Driven Approach*, Yourdon Press, Englewood Cliffs, New Jersey, 1992.
314. Entsminger, *The Tao of Objects: A Beginner's Guide to Object-Oriented Programming*, M&T Books, Redwood City, California, 1990.
315. Eriksson, "A Correct Example of Multiple Inheritance," *SIGPLAN Notices*, Vol. 25, No. 7, July 1990, pp. 7 - 10.
316. Fabry, "Capability Based Addressing," *Communications of the ACM*, Vol. 17, No. 7, July 1974, pp. 403 - 412.
317. Fenton and K. Beck, "Playground: An Object-Oriented Simulation System With Agent Rules for Children of All Ages," *OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices*, Vol. 24, No. 10, October 1989, pp. 123 - 138.

318. Ferber, "Computational Reflection In Class Based Object-Oriented Languages," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 317 - 326.
319. Fichman and C.F. Kemerer, "Object-Oriented and Conventional Analysis and Design Methodologies," IEEE Computer, Vol. 25, No. 10, October 1992, pp. 22 - 39.
320. Fiedler, "Object-Oriented Unit Testing," Hewlett-Packard Journal, Vol. 36, No. 4, April 1989, pp. 69 - 74.
321. Firesmith, "AFATDS: Productivity Improvements Using Ada and Object-Oriented Development," Methodologies and Tools for Real-Time Systems Conference Proceedings, National Institute for Software Quality and Productivity, Washington, D.C., November 1988, pp. L1 - L20.
322. Firesmith, "OOD and Ada Bibliography," Ada Letters, Vol. 10, No. 6, July-August 1990, pp. 114 - 128.
323. Firesmith, "Structured Analysis and Object-Oriented Development Are Not Compatible," Ada Letters, Vol. XI, No. 9, November/December 1991, pp. 56 - 66.
324. Firesmith, "Object-Oriented Graphics for Requirements Analysis and Logical Design," Ada Letters, Vol. XI, No. 9, November/December 1991, pp. 100 - 115.
325. Firesmith, Object-Oriented Requirements Analysis and Logical Design: A Software Engineering Approach, John Wiley and Sons, New York, New York, 1993.
326. Fishman, D. Beech, H.P. Cate, E.C. Chow, T. Connors, J.W. Davis, N. Derrett, C.G. Hoch, W. Kent, P. Lyngbaek, B. Mahbod, M.A. Neimat, T.A. Ryan and M.C. Shan, "Iris: An Object-Oriented Database Management System," ACM Transactions on Office Information Systems, Vol. 5, No. 1, January 1987, pp. 48 - 69.
327. Florentin, Editor, Object-Oriented Programming Systems: Tools and Applications, Chapman and Hall, New York, New York, 1991.
328. Floyd, "A Class Act: Object-Oriented Programming Makes a Case for Software Recycling," Dr. Dobb's Journal of Software Tools, Vol. 14, No. 4, April 1989, pp. 58 - 62.
329. Foote and R. Johnson, "Reflective Facilities in Smalltalk-80," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 327 - 335.
330. Forestier, C. Fornarino, and P. Franchi-Zanettacci, "Ada++: A Class and Inheritance Extension for Ada," Ada: the Design Choice -- Proceedings of the Ada-Europe Conference, Madrid 13-15 June 1989, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 3 - 15.
331. Fors, U. Olsson, and G. Larsson, "The Use of Ada in a Large Shipborne Weapon Control System," in Ada In Industry: Proceedings of the Ada-Europe International Conference Munich 7-9 June, 1988, Cambridge University Press, Cambridge, United Kingdom, 1988, pp. 83 - 93.
332. Franz, Object-Oriented Programming: Featuring Actor, Scott, Foresman and Company, Glenview, Illinois, 1990.

333. Fredericks, "A Small Trellis Programming Project," *Journal of Object-Oriented Programming*, Vol. 5, No. 1, March/April 1992, pp. 12, 14, 16, 18 - 20.
334. Freeman and A.I. Wasserman, *Software Development Methodologies and Ada*, Department of Defense Ada Joint Program Office, 1982.
335. M Freitas, A Moreira, and P. Guerreiro, "Object-Oriented Requirements Analysis in an Ada Project," *Ada Letters*, Vol. 10, No. 6, July-August 1990, pp. 97 - 109.
336. Friedman and M. Wand, "Reification: Reflection Without Metaphysics," *Conference Proceedings of Lisp and Functional Programming*, Association for Computing Machinery, New York, New York, 1984, pp. 348 - 355.
337. Fulbright, "A Comparison Between Functional Decomposition and Object-Oriented Methodologies -- A Case Study," *Proceedings of the Eighth Annual National Conference on Ada Technology*, March 5-8, 1990, pp. 299 - 306.
338. Gabriel, "Using the Common LISP Object System," *Computer Language*, Vol. 6, No. 8, August 1989, pp. 73 - 80.
339. Gallo, R. Minot, and I. Thomas, "The Object Management System of PCTE as a Software Engineering Database Management System," *SIGPLAN Notices*, Vol. 22, No. 1, January 1987, pp. 12 - 15.
340. Gallo, G. Serrano, and F. Tisato, "ObNet: An Object-Oriented Approach for Supporting Large, Long-Lived, Highly-Configurable Systems," *Proceedings of the 11th International Conference on Software Engineering*, May 15-18, 1989, pp. 138 - 144.
341. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1995.
342. Gardner, "Successes and Limitations of Object-Oriented Design," *Journal of Pascal, Ada, and Modula-2*, Vol. 7, No. 6, November/December, 1988, pp. 30 - 41.
343. Garfunkel, "Cobol -- The Next Stage," *ComputerWorld*, Vol. XXIV, No. 30, July 23, 1990, pp. 87 - 91.
344. Genolini, A. Di Maio, C. Cardigno, S. Goldsack, and C. Atkinson, "Specifying Synchronization Constraints In a Concurrent Object-Oriented Language," *Technology of Object-Oriented Languages and Systems 1989 (TOOLS '89)*, Paris, France, November 13 - 15, 1989.
345. Genolini, A. Di Maio, and M. De Michele, "DRAGOON and Ada: the Wedding of the Nineties," *Proceedings of the Seventh Washington Ada Symposium*, June 25-28, 1990, pp. 245 - 254.
346. George and J. Sodhi, "Objects With Multiple Representations in Ada," *Proceedings of the Seventh Annual National Conference on Ada Technology*, March 13-16, 1989, pp. 567 - 575.
347. Geschke, J. Morris, and E. Satterthwaite, "Early Experience With Mesa," *Communications of the ACM*, Vol. 20, No. 8, August 1977, pp. 540 - 553.

348. Ghezzi, D. Mandrioli, S. Morasca, and M. Pezze, "A General Way to Put Time in Petri Nets," Proceedings of the Fifth International Workshop On Software Specification and Design, May 19-20, 1989, Pittsburgh, Pennsylvania, IEEE Computer Society Press, Washington, D.C., May 1989, pp. 60 - 67.
349. Ghezzi, M. Jazayeri, and D. Mandrioli, Fundamentals of Software Engineering, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
350. Gibbs, D. Tsichritzis, E. Casais, O. Nierstrasz and X. Pintado, "Class Management for Software Communities," Communications of the ACM, Vol. 33, No. 9, September 1990, pp. 90 - 103.
351. Gibson, "Objects -- Born and Bred," Byte, Vol. 15, No. 10, October 1990, pp. 245 - 246, 248, 250, 252, 254.
352. Gibson, "Flattening the Learning Curve: Educating Object-Oriented Developers," Journal of Object-Oriented Programming, Vol. 3, No. 6, February 1991, pp. 24 - 29.
353. Gill, "MIS Slowly Warms Up to Object-Oriented Programming," ComputerWorld, Vol. XXII, No. 8, February 22, 1988, pp. 71 - 76.
354. Gindre and F. Sada, "A Development in Eiffel: Design and Implementation of a Network Simulator," Journal of Object-Oriented Programming, Vol. 2, No. 1, May/June 1989, pp. 27 - 33.
355. Glassey and S. Adiga, "Conceptual Design of a Software Object Library for Simulation of Semiconductor Manufacturing Systems," Journal of Object-Oriented Programming, Vol. 2, No. 4, September 1989, pp. 39 - 43.
356. Goguen and J. Meseguer, "Unifying Functional, Object-Oriented and Relational Programming With Logical Semantics," in Research Directions in Object-Oriented Programming, B. Shriver and P. Wegner, Editors, The MIT Press, Cambridge, Massachusetts, 1987, pp. 417 - 477.
357. Goldberg and A. Kay, Editors, Smalltalk-72 Instructional Manual, Technical Report SSL-76-6, Xerox PARC, Palo Alto, California, March 1976.
358. Goldberg and D. Robson, Smalltalk-80: The Language and Its Implementation, Addison-Wesley, Reading, Massachusetts, 1983.
359. Goldberg, Smalltalk-80: The Interactive Programming Environment, Addison-Wesley, Reading, Massachusetts, 1984.
360. Goldsack and C. Atkinson, "An Object Oriented Approach to Virtual Nodes: Are Package Types an Answer?," Proceedings of the Third International Workshop on Real-Time Ada Issues (special issue of Ada Letters), Vol. 10, No. 4, Spring 1990, pp. 78 - 84.
361. Goldsack, Ada for Specification : Possibilities and Limitations, Cambridge University Press, Cambridge, United Kingdom, 1985.
362. Gomaa, "A Software Design Method for Real-Time Systems," Communications of the ACM, Vol. 27, No. 9, September 1984, pp. 938 - 949.

363. Goodenough, "Exception Handling: Issues and a Proposed Notation," *Communications of the ACM*, Vol. 18, No. 12, pp. 683 - 696.
364. Gopinath, T. Bihari, and R. Gupta, "Compiler Support for Object-Oriented Real-Time Software," *IEEE Software*, Vol. 9, No. 5, September 1992, pp. 45 - 50.
365. Gorlen, S.M. Orlow, and P.S. Plexico, *Data Abstraction and Object-Oriented Programming in C++*, John Wiley & Sons, New York, New York, 1990.
366. Gorman and J. Choobineh, "The Object-Oriented Entity-Relationship Model," *Journal of Management Information Systems*, Vol. 7, No. 3, Winter 1991. pp. 41 - 65.
367. Graham, *Object-Oriented Methods*, Addison-Wesley, Reading, Massachusetts, 1991.
368. Graham, "Migration Using SOMA: A Semantically Rich Method of Object-Oriented Analysis," *Journal of Object-Oriented Programming*, Vol. 5, No. 9, February 1993, pp. 31 - 42.
369. Graham, *Migrating to Object Technology*, Addison-Wesley, Reading, Massachusetts, 1995.
370. Graube, "Reflexive Architecture: From ObjVLisp to CLOS," *ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science*, Volume 322, 1988, pp. 110 - 127.
371. Graube, "Metaclass Compatibility," *OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices*, Vol. 24, No. 10, October 1989, pp. 305 - 315.
372. Gray, "Procedures for Transitioning from Structured Methods to Object-Oriented Design," *Proceedings of the Conference on Methodologies and Tools for Real-Time Systems IV*, National Institute for Software Quality and Productivity, Washington, D.C., September 14-15 1987, pp. R-1 to R-21.
373. Gray, "Transitioning from Structured Analysis to Object-Oriented Design," *Proceedings of the Fifth Washington Ada Symposium*, June 27 - 30, 1988, Association for Computing Machinery, New York, New York, 1988, pp. 151 - 162.
374. Grogono and A. Bennett, "Polymorphism and Type Checking in Object-Oriented Languages," *SIGPLAN Notices*, Vol. 24, No. 11, November 1989, pp. 109 - 115.
375. Guilfoyle, R. Pirchner, L. Von Gerichten, M. Ginsberg, and D. Clarson, "A Comparison of Methods Which Address the Development of Real-Time Embedded Systems," *Proceedings of the Seventh Annual National Conference on Ada Technology*, March 13-16, 1989, pp. 67 - 77.
376. Gupta and E. Horowitz, Editors, *Object-Oriented Databases With Applications to CASE, Networks, and VLSI CAD*, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
377. Guttag, "Abstract Data Types and the Development of Data Structures," *Communications of the ACM*, Vol. 20, No. 6, June 1977, pp. 396 - 404.
378. Gwinn, "Object-Oriented Programs in Realtime," *SIGPLAN Notices*, Vol. 27, No. 2, February 1992, pp. 47 - 56.

379. Hailpern and H. Ossher, "Extending Objects to Support Multiple Interfaces and Access Control," *IEEE Transactions on Software Engineering*, Vol. 16, No. 11, November 1990, pp. 1247 - 1257.
380. Halbert and P.D. O'Brien, "Using Types and Inheritance in Object-Oriented Programming," *IEEE Software*, Vol. 4, No. 5, September 1987, pp. 71 - 79.
381. Hamilton, "IBM On Object-Oriented Path," *ComputerWorld*, Vol. XXIV, No. 44, November 5, 1990, pp. 1, 6.
382. Hammer and D. McLeod, "Database Description with SMD: A Semantic Data Model," *ACM Transactions on Database Systems*, Vol. 6, No. 3, September 1981, pp. 351 - 386.
383. Hanson, *Data Structure Program Design*, Ken Orr and Associates, Topeka, Kansas, 1983.
384. Harding, "Standards Group Clears Way for OO Cobol," *Software*, Vol. 11, No. 3, March 1991, page 45.
385. Harel, A. Pnueli, J.P. Schmidt, and R. Sherman, "On the Formal Semantics of Statecharts," *Proceedings of the Second IEEE Symposium on the Logic of Computer Science*, 1987, pp. 54 - 64.
386. Harel, "Statecharts: A Visual Formalism for Complex Systems," *Science of Computer Programming*, Vol. 8, No. 3, June 1, 1987, pp. 231 - 274.
387. Harland, *Polymorphic Programming Languages -- Design and Implementation*, Halstead Press, New York, New York, 1984.
388. Harmon, "Object-Oriented Systems," *Intelligent Software Strategies*, Vol. 6, No. 9, pp. 1 - 16.
389. Hatley and I.A. Pirbhai, *Strategies for Real-Time System Specification*, Dorset House Publishing, New York, New York, 1988.
390. Hayes and D. Coleman, "Coherent Models for Object-Oriented Analysis," *OOPSLA '91 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 26, No. 11, November 1991, pp. 171 - 183.
391. Heitz and B. Labreuille, "Design and Development of Distributed Software Using Hierarchical Object Oriented Design and Ada," in *Ada In Industry: Proceedings of the Ada-Europe International Conference Munich 7-9 June, 1988*, Cambridge University Press, Cambridge, United Kingdom, 1988, pp. 143 - 156.
392. Heitz, "HOOD: A Hierarchical Object-Oriented Design Method," *Proceedings of the Third German Ada Users Congress*, January 1988, Gesellschaft für Software Engineering, Munich, West Germany, pp. 12-1 - 12-9.
393. Hekmatpour, *C++: A Guide for C Programmers*, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
394. Helm, I.M. Holland, and D. Gangopadhyay, "Contracts: Specifying Behavioral Compositions in Object-Oriented Systems," *OOPSLA/ECOOP '90 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 25, No. 10, October 1990, pp. 169 - 180.

395. Henderson-Sellers and L.L. Constantine, "Object-Oriented Development and Functional Decomposition, Journal of Object-Oriented Programming, Vol. 3, No. 5, January/February 1991, pp. 11 - 17.
396. Henderson-Sellers and J.M. Edwards, "The Object-Oriented Systems Life-Cycle," Communications of the ACM, Vol. 33, No. 9, September 1990, pp. 145 - 159.
397. Henderson-Sellers, "Hybrid Object-Oriented/Functional Decomposition Methodologies for the Software Engineering Lifecycle," Hotline on Object-Oriented Technology, Vol. 2, No. 7, May 1991, pp. 1, 2 - 8.
398. Henderson-Sellers, A Book of Object-Oriented Knowledge -- Object-Oriented Analysis, Design and Implementation: A New Approach to Software Engineering, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
399. Henderson-Sellers and J.M. Edwards, "Identifying Three Levels of O-O Methodologies," Report on Object Analysis and Design, Vol. 1, No. 2, July-August 1994, pp. 25 - 28.
400. Henderson-Sellers and J.M. Edwards, BOOKTWO of Object-Oriented Knowledge: The Working Object, Prentice Hall, Englewood Cliffs, New Jersey, 1994.
401. Hendler, "Enhancement for Object-Oriented Programming," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 98 - 106.
402. Hewitt, P. Bishop, and R. Steiger, "A Universal Modular ACTOR Formalism for Artificial Intelligence," Proceedings of the Third International Joint Conference on Artificial Intelligence, August 1973, pp. 235 - 245.
403. Hewitt, T. Reinhardt, G. Agha, and G. Attardi, "Linguistic Support for Receptionists for Shared Resources," in Semantics of Concurrency, Papers of the Seminar, Lecture Notes on Computer Science, Vol. 197, S.D. Brookes, A.W. Roscoe, and G. Winskel, Editors, Springer-Verlag, New York, New York, 1985, pp. 330 - 359.
404. Hewitt, "Viewing Control Structures As Patterns of Passing Messages," Artificial Intelligence, Vol. 8, 1977, pp. 323 - 364.
405. Hibbard, A. Hisgen, J. Rosenberg, M. Shaw, and M. Sherman, Studies in Ada Style, 2nd. Edition, Springer-Verlag, New York, New York, 1983.
406. Higa, M. Morrison, J. Morrison, and O.R.L. Sheng, "An Object-Oriented Methodology for Knowledge Base / Database Coupling," Communications of the ACM, Vol. 35, No. 6, June 1992, pp. 99 - 113.
407. Hirasuna, "Using Inheritance and Polymorphism With Ada in Government Sponsored Contracts," Ada Letters, Vol. 12, No. 2, March/April 1992, pp. 43 - 56.
408. Hoare, "Monitors: An Operating System Structuring Concept," Communications of the ACM, Vol. 17, No. 10, October 1974, pp. 549 - 557.
409. Hoare, "Communicating Sequential Processes," Communications of the ACM, Vol. 21, No. 8, August 1978, pp. 666 - 677.
410. Hoare, Communicating Sequential Processes, Prentice Hall, Englewood Cliffs, New Jersey, 1985.

411. Hoffman, "On Criteria for Module Interfaces," IEEE Transactions on Software Engineering, Vol. 16, No. 5, May 1990, pp. 537 - 542.
412. Honda and A. Yonezawa, "Debugging Concurrent Systems Based on Object Groups," in ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, S. Gjessing and K. Nygaard, Editors, Springer Verlag, New York, New York, 1988, pp. 267 - 282
413. Horn, "Conformance, Genericity, Inheritance, and Enhancement," ECOOP '87: Proceedings of the European Conference on Object-Oriented Programming, Lecture Notes on Computer Science, Volume 276, Springer Verlag, New York, New York, 1987, pp. 223 - 233.
414. Horowitz and R.C. Williamson, "SODOS: A Software Documentation Support Environment -- Its Use," IEEE Transactions on Software Engineering, Vol. SE-12, No. 11, November 1986, pp. 1076 - 1087.
415. Horowitz, Fundamentals of Programming Languages, Computer Science Press, Rockville, Maryland, 1983.
416. "Towards An Object-Oriented Method for System Architecture Design," Proceedings of the 1990 IEEE International Conference on Computer Systems and Software Engineering -- EuroComp '90, Tel-Aviv, Israel, May 8 - 10, 1990, pp. 12 - 17.
417. Hudson and R. King, "The Cactis Project: Database Support for Software Environments," IEEE Transactions on Software Engineering, Vol. 14, No. 6, June 1988, pp. 709 - 719.
418. Hufnagel and J.C. Brown, "Performance Properties of Vertically Partitioned Object-Oriented Systems," IEEE Transactions on Software Engineering, Vol. 15, No. 8, August 1989, pp. 935 - 946.
419. Hughes, Object-Oriented Databases, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
420. Hull and R. King, "Semantic Database Modeling: Survey, Applications, and Research Issues," ACM Computing Surveys, Vol. 19, No. 3, September 1987, pp. 201 - 260.
421. Hull, A. Zarea-Aliabadi, and D. Gutherie, "Object-Oriented Design, Jackson System Development (JSD) Specifications and Concurrency," Software Engineering Journal, Vol. 4, No. 2, March 1989, pp. 79 - 86.
422. Ibrahim, W.E. Bejcek, and F.A. Cummins, "Instance Specialization Without Delegation," Journal of Object-Oriented Programming, Vol. 4, No. 3, June 1991, pp. 53 - 56.
423. International Business Machines Corporation, Object-Oriented Analysis of the ITSO Common Scenario, IBM Corporation, International Technical Support Center, Research Triangle Park, North Carolina, 1990.
424. IEEE, IEEE Standard Glossary of Software Engineering Terminology, The Institute of Electrical and Electronic Engineers, New York, New York, 1983.
425. IEEE, Special Issue on Object-Oriented Systems, IEEE Database Engineering, Vol. 8, No. 4, December 1985.

426. Ince, Object-Oriented Software Engineering With C++, McGraw-Hill Book Company, New York, New York, 1991.
427. Ingalls, "The Smalltalk-76 Programming System Design and Implementation," Fifth Annual ACM Symposium on the Principles of Programming Languages, January 1978, pp. 9 - 15.
428. Ingrassia, "The Unit Development Folder (UDF): An Effective Management Tool for Software Development," TRW Technical Report TRW-SS-76-11, 1976. reprinted in IEEE Tutorial: Software Management, Second Edition, edited by D.J. Reifer, IEEE Catalog Number EHO189-1, Computer Society Order Number 396.
429. Intel Corporation, iAPX 432 Object Primer, Manual 171858-001 Rev. B, Aloha, Oregon, 1980.
430. Ishikawa, "Reflection Facilities and Realistic Programming," SIGPLAN Notices, Vol. 26, No. 8, August 1991, pp. 101 - 110.
431. Ishikawa, H. Tokuda, and C.W. Mercer, "An Object-Oriented Real-Time Programming Language," IEEE Computer, Vol. 25, No. 10, October 1992, pp. 66 - 73.
432. Jackson, System Development, Prentice Hall, Englewood Cliffs, New Jersey, 1983.
433. Jackson, "Developing Ada Programs Using the Vienna Development Method (VDM)," Software Practice and Experience, Vol. 15, No. 3, March 1985, pp. 305 - 318.
434. Jackson, Introduction to Expert Systems, Addison-Wesley, Reading, Massachusetts, 1986.
435. Jacky and I.J. Kalet, "An Object-Oriented Programming Mechanism for Standard Pascal," Communications of the ACM, Vol. 30, No. 9, September 1987, pp. 772 - 776.
436. Jacobson, "Language Support for Changeable Large Real Time Systems," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 377 - 384.
437. Jacobson, "Object-Oriented Development In an Industrial Environment," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 183 - 191.
438. Jacobson, M. Christerson, P. Jonsson, and G. Övergaard, Object-Oriented Software Engineering: A Use Case Driven Approach, Addison-Wesley, Reading, Massachusetts, 1992.
439. Jacobson, M. Ericsson, and A. Jacobson, The Object Advantage: Business Process Reengineering With Object Technology, Addison-Wesley, Reading, Massachusetts, 1995.
440. Jalote, "Resilient Objects in Broadcast Networks," IEEE Transactions on Software Engineering, Vol. 15, No. 1, January 1989, pp. 68 - 72.
441. Jalote, "Functional Refinement and Nested Objects for Object-Oriented Design," IEEE Transactions on Software Engineering, Vol. 15, No. 3, March 1989, pp. 264 - 270.
442. Jamsa, "Object Oriented Design vs. Structured Design -- A Student's Perspective," Software Engineering Notes, Vol. 9. No. 1, January 1984, pp. 43 - 49.

443. Janlert, "Pictorial Knowledge Representation," Proceedings of the European Conference on Artificial Intelligence 1988, Munich, August 1988, pp. 149 - 151.
444. Jaworski and D. LaVallee, "Principles for Defining an Object-Oriented Design Decomposition in Ada," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 173 - 182.
445. Jean and A. Strohmeier, "An Experience In Teaching OOD for Ada Software," Software Engineering Notes, Vol. 15., No. 5, October 1990, pp. 44 - 49.
446. Jellinghaus, "Eiffel Linda: An Object-Oriented Linda Dialect," ACM SIGPLAN Notices, Vol. 25, No. 12, December 1990, pp. 70 - 84.
447. Jensen and C.C. Tonies, Software Engineering, Prentice Hall, Englewood Cliffs, New Jersey, 1979.
448. Jochem, M. Rabe, W. Süssenguth, and P. Bals, "An Object-Oriented Analysis and Design Methodology for Computer Integrated Manufacturing Systems," Technology of Object-Oriented Languages and Systems 1989 (TOOLS '89), Paris, France, November 13 - 15, 1989, pp. 75 - 84.
449. Johnson and B. Foote, "Designing Reusable Classes," Journal of Object-Oriented Programming, Vol. 1, No. 2, July/August 1988, pp. 22-35.
450. Johnson and B. Foote, "Reflective Facilities in Smalltalk-80," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 327 - 335.
451. Johnson, J.O. Graver, and L.W. Zurawski, "TS: An Optimizing Compiler for Smalltalk," OOPSLA '88 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 18 - 26.
452. Johnson, "Type-Checking Smalltalk," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 315-321.
453. Jones, R.E. Bozman, and W. McIver, "The Morehouse Object-Oriented Reuse Library System," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 456 - 462.
454. Jones, Software Development: A Rigorous Approach, Prentice Hall, Englewood Cliffs, New Jersey, 1980.
455. Jones, Editor, Tutorial: Programmer Productivity: Issues for The Eighties, IEEE Catalog Number EHO186-7, Computer Society Order Number 391, Computer Society Press of the IEEE, Washington, D.C., 1981.
456. Jones, Systematic Software Development Using VDM, Prentice Hall, Englewood Cliffs, New Jersey, 1986.
457. Jordan, "Implementation Benefits of C++ Language Mechanisms," Communications of the ACM, Vol. 33, No. 9, September 1990, pp. 61 - 64.
458. Jordan, R. Similan, and A. Wilkinson, "Streamlining the Project Cycle With Object-Oriented Requirements," Proceedings of the Ninth Annual Conference on Object-Oriented

Programming Systems, Languages, and Applications, ACM SIGPLAN Notices, Vol. 29, No. 10, October 1994, pp. 287 - 300.

- 459. Kafura and K.H. Lee, "Inheritance in Actor Based Concurrent Object-Oriented Languages," ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 131 - 145.
- 460. Kahn, E. Tribble, M. Miller, and D. Bobrow, "Vulcan: Logical Concurrent Objects," in Research Directions in Object-Oriented Programming, B. Shriver and P. Wegner, Editors, The MIT Press, Cambridge, Massachusetts, 1987, pp. 75 - 112.
- 461. Kaiser and D. Galen, "Melding Data Flow and Object-Oriented Programming," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 254 - 267.
- 462. Kaiser and B. Hailpern, "An Object-Based Programming Model for Shared Data," ACM Transactions on Programming Languages and Systems, Vol. 14, No. 2, April 1992, pp. 201 - 264.
- 463. Kamin, "Inheritance in Smalltalk-80: A Denotational Definition," Proceedings of the Conference on Principles of Programming Languages, 1988, pp. 80 -87.
- 464. Katz and E. Chang, "Managing Change in a Computer-Aided Design Database," Proceedings of the 13th International Conference on Very Large Data Bases, Brighton, England, September 1987, pp. 455 - 462.
- 465. Kavi and D.J. Chen, "Architectural Support for Object-Oriented Languages," Digest of Papers COMPCON, Spring 1987, IEEE Catalog Number 87CH2409-1, Computer Society Order Number 764, Computer Society Press of the IEEE, Washington, D.C., pp. 54 - 58.
- 466. Kay, Flex: A Flexible-Extensible Language, Masters Thesis, Department of Computer Science, University of Utah, 1968.
- 467. Kay, The Reactive Engine, Ph.D. Thesis, Department of Computer Science, University of Utah, August 1969.
- 468. Kay, "Microelectronics and the Personal Computer," Scientific American, Vol. 237, No. 3, September 1977, pp. 230 - 244.
- 469. Kay, "The Early History of Smalltalk," SIGPLAN Notices, Vol. 28, No. 3, March 1993, pp. 69 - 95.
- 470. Keene, Object-Oriented Programming in Common Lisp, Addison-Wesley, Reading, Massachusetts, 1989.
- 471. Kelly, "A Comparison of Four Design Methods for Real-Time Systems," Proceedings of the 9th International Conference on Software Engineering, March 30-April 2, 1987, pp. 238 - 252.
- 472. Kelly and Y.S. Sherif, "Comparison Four Design Methods For Real-Time Software Development," Information and Software Technology, Vol. 34, No. 2, February 1992, pp. 74 - 82.

473. Kent, "A Rigorous Model of Object Reference, Identity, and Existence," *Journal of Object-Oriented Programming*, Vol. 4, No. 3, June 1991, pp. 28 - 34, 36.
474. Kernighan and P.J. Plauger, *The Elements of Programming Style*, 2nd Ed., McGraw-Hill Book Company, New York, New York, 1978.
475. Ketabchi and R. Weins, "Implementation of Persistent Multi-User Object-Oriented Systems," *Digest of Papers COMPCON*, Spring 1987, IEEE Catalog Number 87CH2409-1, Computer Society Order Number 764, Computer Society Press of the IEEE, Washington, D.C., pp. 44 - 49.
476. Keyes, "Unstructured Objects Focus of New Analysis," *Software*, Vol. 11, No. 3, March 1991, pp. 40 - 42, 46 - 48, 50.
477. Keyes, "Code Trapped Between Legacy, Object Worlds," *Software*, Vol. 12, No. 8, June 1992, pp. 39 - 41, 44, 46.
478. Khalsa, "Using Object Modeling to Transform Structured Analysis Into Object-Oriented Design," *Proceedings of the Sixth Washington Ada Symposium*, June 26-29, 1989, pp. 201 - 212.
479. Khoshafian and R. Abnous, *Object Orientation: Concepts, Languages, Databases, User Interfaces*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1990.
480. Khoshafian and G.P. Copeland, "Object Identity," *OOPSLA '86 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 21, No. 11, November 1986, pp. 406 - 416.
481. Kilian, "Trellis: Turning Designs Into Programs," *Communications of the ACM*, Vol. 33, No. 9, September 1990, pp. 65 - 67.
482. Kilov, "Object Concepts and Bibliography," *ACM SIGPLAN Notices*, Vol. 26, No. 10, October 1991, pp. 11 - 12.
483. Kim and F. Lochovsky, *Object-Oriented Concepts, Databases, and Applications*, ACM Press/Addison Wesley, Reading, Massachusetts, 1989.
484. Kim, J. Banerjee, H.-T. Chou, J.F. Garza, and D. Woelk, "Composite Object Support in an Object-Oriented Database System," *OOPSLA '87 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 22, No. 12, December 1987, pp. 118-125.
485. Kim, E. Bertino, and J.F. Garza, "Composite Objects Revisited," *Proceedings of the ACM SIGMOD '89 Conference*, pp. 337 - 347.
486. Kim, J.-M. Nicolas, and S. Nishio, Editors, *Deductive and Object-Oriented Database: Proceedings of the First International Conference on Deductive and Object-Oriented Databases (DOOD 89) Kyoto Research Park, Kyoto, Japan, 4-6 December, 1989*, North-Holland (Elsevier), New York, New York, 1990.
487. Kim, *Introduction to Object-Oriented Databases*, The MIT Press, Cambridge, Massachusetts, 1990.
488. Kirkerud, *Object-Oriented Programming With SIMULA*, Addison-Wesley, Reading, Massachusetts 1989.

489. Knudsen and O.L. Madsen, "Teaching Object-Oriented Programming Is More Than Teaching Object-Oriented Programming Languages," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 21 - 40.
490. Knudsen, "Name Collision In Multiple Classification Hierarchies," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 93 - 109.
491. Knuth, The Art of Computer Programming, Volume 1/Fundamental Algorithms, 2nd Ed., Addison-Wesley, Reading, Massachusetts, 1973.
492. Koenig and B. Stroustrup, "Exception Handling for C++," Journal of Object-Oriented Programming, Vol. 3, No. 2, July/August 1990, pp. 16 - 33.
493. Korienek and T. Wrensch, A Quick Trip to Objectland, Prentice Hall, Englewood Cliffs, New Jersey, 1995.
494. Korson and J.D. McGregor, "Understanding Object-Oriented: A Unifying Paradigm," Communications of the ACM, Vol. 33, No. 9, September 1990, pp. 40 - 60.
495. Korson and V.K. Vaishnavi, "Managing Emerging Software Technologies: A Technology Transfer Framework," Communications of the ACM, Vol. 35, No. 9, September 1992, pp. 101 - 111.
496. Korson, V. Vaishnavi, and B. Meyer, Editors, Technology of Object-Oriented Languages and Systems: Tools 5, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
497. Koschmann and M.W.Evens, "Bridging the Gap Between Object-Oriented and Logic Programming," IEEE Software, Vol. 5, No. 4, July 1988, pp. 36 - 42.
498. Kovarik and S. Nies, "Supporting Object-Oriented Programming Within Ada: Extending the Paradigm," Proceedings of the Fourth Annual Conference on Artificial Intelligence and Ada, November 1988, pp. 13-1 - 13-6.
499. Krasner, Smalltalk-80: Bits of History, Words of Advice, Addison-Wesley, Reading, Massachusetts, 1983.
500. Kristensen, O.L. Madsen, B. Moller-Pedersen and K. Nygaard, "The BETA Programming Language," in Research Directions in Object-Oriented Programming, B. Shriver and P. Wegner, Editors, The MIT Press, Cambridge, Massachusetts, 1987, pp. 7 - 48.
501. Krogdahl and K.A. Olsen, "Ada As Seen From Simula," Software Practice and Experience, Vol. 16, No. 8, August 1986, pp. 687 - 700.
502. Kruchten, "Error Handling in Large, Object-Based Ada Systems," Ada Letters, Vol. X, No. 7, September/October 1990, pp. 91 - 103.
503. Kuhn, The Structure of Scientific Revolutions, University of Chicago Press, Chicago, Illinois, 1962.
504. Kull, "Object: Build a Better System," Computer Decisions, Vol. 15, No. 25, November 4, 1986, pp. 38 - 41.

- 505. Kung, "Object Subclass Hierarchy in SQL: A Simple Approach," Communications of the ACM, Vol. 33, No. 7, July 1990, pp. 117 - 125.
- 506. Ladd, "The Art of Abstraction," Computer Language, Vol. 6, No. 11, November 1989, pp. 137 - 140.
- 507. Ladd, C++ Techniques and Applications, M&T Books, Redwood City, California, 1990.
- 508. Ladden, "A Survey of Issues To Be Considered In the Development of an Object-Oriented Development Methodology for Ada," Software Engineering Notes, Vol. 13, No. 3, July 1988, pp. 24 - 31.
- 509. LaLonde and J.R. Pugh, Inside Smalltalk, Volume 1, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
- 510. LaLonde, D.A. Thomas, and J.R. Pugh, "An Exemplar Based Smalltalk," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 322 - 330.
- 511. Lamb, "Specification of Iterators," IEEE Transactions on Software Engineering, Vol. 16, No. 12, December 1990, pp. 1352 - 1360.
- 512. Laranjeira, "Software Size Estimation of Object-Oriented Systems," IEEE Transactions on Software Engineering, Vol. 16, No. 5, May 1990, pp. 510 - 522.
- 513. Lawless and M.M. Miller, Understanding CLOS: The Common Lisp Object System, Digital Press, Bedford, Massachusetts, 1990.
- 514. Ledbetter and B. Cox, "Software ICs," Byte, Vol. 10, No. 6, June 1985, pp. 307 - 315.
- 515. Ledgard and M. Marcotty, The Programming Language Landscape: Syntax, Semantics, Implementation, Second Edition, Science Research Associates, Chicago, Illinois, 1986.
- 516. Lee and D.L. Carver, "Object-Oriented Analysis and Specification: A Knowledge Base Approach," Journal of Object-Oriented Programming, Vol. 3, No. 5, January/February 1991, pp. 35 - 43.
- 517. Lee and M.S. Rissman, An Object-Oriented Solution Example: A Flight Simulator Electrical System, Technical Report CMU/SEI-89-TR-5, Software Engineering Institute, Pittsburgh, Pennsylvania, 1989.
- 518. Lee, M.S. Rissman, R.D. D'Ippolito, C. Plinta, and R. Van Scoy, An OOD Paradigm for Flight Simulators, Technical Report CMU/SEI-87-TR-43 (ESD-TR-87-206), Software Engineering Institute, Pittsburgh, Pennsylvania, 1987.
- 519. Leavens and W.E. Weihl, "Reasoning About Object-Oriented Programs That Use Subtypes," OOPSLA/ECOOP '90 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990, pp. 212 - 223.
- 520. Leveson, "Software Safety: What, Why, and How," ACM Computing Surveys, Vol. 18, No. 2, June 1986, pp. 125 - 164.
- 521. Levy, Capability-Based Computer Systems, Digital Press, Bedford, Massachusetts, 1984.

522. Lewis, S.M. Henry, D.G. Kafura, and R.S. Schulman, "An Empirical Study of the Object-Oriented Paradigm and Software Reuse," OOPSLA '91 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 26, No. 11, November 1991, pp. 184 - 196.
523. Lewis, "Object-Oriented Programming A Hit," IEEE Spectrum, Vol. 27, No. 1, January 1990, pp. 38 - 39.
524. Li and D. McLeod, "Object Flavor Evolution in an Object-Oriented Database System," Proceedings of the Conference on Office Information Systems, edited by Robert B. Allen, March 1988, pp. 265 - 275.
525. Li, "Integration of Structured and Object-Oriented Programming," in Focus On Analysis and Design, SIGS Publications, Inc., New York, New York, 1991, pp. 54 - 60.
526. Lieberherr and I.M. Holland, "Assuring Good Style for Object-Oriented Programs," IEEE Software, Vol. 6, No. 5, September 1989, pp. 38 - 48.
527. Lieberherr and A.J. Riel, "Demeter: A Case Study of Software Growth Through Parameterized Classes," Proceedings of the 10th International Conference on Software Engineering, April 11-15, 1988, pp. 254 - 264.
528. Lieberherr and A.J. Riel, "Demeter: a CASE Study of Software Growth Through Parameterized Classes," Journal of Object-Oriented Programming, Vol. 1, No. 3, August/September 1988, pp. 8 - 22.
529. Lieberherr and A.J. Riel, "Contributions to Teaching Object-Oriented Design and Programming," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 11 - 22.
530. Lieberherr, I. Holland, and A.J. Riel, "Object-Oriented Programming: An Objective Sense of Style," OOPSLA '88 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 323 - 334.
531. Lieberherr, "Object-Oriented Programming With Class Dictionaries," Journal of LISP and Symbolic Computation, Vol. 1, No. 2, 1988, pp. 185 - 212.
532. Lieberman and C. Hewitt, "A Real-Time Garbage Collector Based on the Lifetimes of Objects," Communications of the ACM, Vol. 26, No. 6, June 1983, pp. 419 - 429.
533. Lieberman, L.A. Stein, and D. Ungar, "Of Types and Prototypes: The Treaty of Orlando," OOPSLA '87 Addendum to the Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 5, May 1988, pp. 43 - 44.
534. Lieberman, "Using Prototypical Objects to Implement Shared Behavior In Object-Oriented Systems," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 214 - 223.
535. Lin and S. Harous, "Reusability and Extensibility in Ada," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 269 - 274.
536. Lipkie, S.R. Evans, J.K. Newlin, and R.L. Weissman, "Star Graphics: An Object-Oriented Implementation," Computer Graphics, Vol. 16, No. 3, July 1982, pp. 115 - 124.

- 537. Lippman, C++ Primer, Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.
- 538. Lippman, C++ Primer, Second Edition, Addison-Wesley Publishing Company, Reading, Massachusetts, 1991.
- 539. Liskov and J. Guttag, Abstraction and Specification in Program Development, The MIT Press, Cambridge, Massachusetts, 1986.
- 540. Liskov and S.N. Zilles, "Specification Techniques for Data Abstraction," IEEE Transactions on Software Engineering, Vol. SE-1, No. 1, March 1975, pp. 7 - 19.
- 541. Liskov and S.N. Zilles, "An Introduction to Formal Specifications of Data Abstraction," in Current Trends In Programming Methodology, R.T. Yeh, Editor, Prentice Hall, Englewood Cliffs, New Jersey, 1977, pp. 1 - 32.
- 542. Liskov, A. Snyder, R. Atkinson, and C. Schaffert, "Abstraction Mechanisms in CLU," Communications of the ACM, Vol. 20, No. 8, August 1977, pp. 564 - 576.
- 543. Liskov, R. Atkinson, T. Bloom, E. Moss, J.C. Schaffert, R. Scheifler, and A. Snyder, CLU Reference Manual, Springer-Verlag, New York, New York, 1981.
- 544. Liskov, "A Design Methodology for Reliable Software Systems," Proceedings, Fall Joint Computer Conference (1972), AFIPS Press, Reston, Virginia, 1972.
- 545. Liskov, "Data Abstraction and Hierarchy," OOPSLA '87 Addendum to the Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 5, May 1988, pp. 17 - 34.
- 546. Liu, "On the Object-Orientedness of C++," SIGPLAN Notices, Vol. 26, No. 3, March 1991, pp. 63 - 67.
- 547. Loomis, A.V. Shaw, and J.E. Raumbaugh, "An Object Modeling Technique for Conceptual Design," Proceedings of ECOOP '87: European Conference on Object-Oriented Programming, Springer Verlag, New York, New York, 1987, pp. 192 - 202.
- 548. Lorenz, Object-Oriented Software Development: A Practical Guide, Prentice Hall, Englewood Cliffs, New Jersey, 1993.
- 549. Love, "Experiences with Smalltalk-80(TM) for Application Development," in IEEE Proceeding of the Softfair: A Conference on Software Development Tools, Techniques, and Alternatives, July 25-28, 1983 pp. 61-65.
- 550. Low, "A Shared Persistent Object Store," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 390 - 410.
- 551. Loy, "Comparisons of O-O and Structured Development," Software Engineering Notes, Vol. 15, No. 1, January 1990, pp. 44 - 48.
- 552. Lubinski and I. Hutzel, "An Object-Oriented Graphical Kernel System," Computer Graphics World, July 1984, pp. 69 - 75.
- 553. Lujun and S. Zhongxiu, "An Object-Oriented Programming Language for Developing Distributed Software," SIGPLAN Notices, Vol. 22, No. 8, August 1987, pp. 51 - 56.

554. Lukman, "Transforming the 2167A Requirements Definition Model Into an Ada-Object-Oriented Design," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 200 - 205.
555. Maciaszek, "AD/Cycle Repository Manager for Object-Oriented Perspective," Software Engineering Notes, Vol. 16, No. 1, January 1991, pp. 50 - 53.
556. MacLennan, "Values and Objects in Programming Languages," SIGPLAN Notices, Vol. 17, No. 12, December 1982, pp. 70 - 79.
557. Madsen and B. Møller-Pedersen, "What Object-Oriented Programming May Be -- and What It Does Not Have to Be," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 1 - 20.
558. Madsen, B. Magnusson, and B. Møller-Pedersen, "Strong Typing of Object-Oriented Languages Revisited," OOPSLA/ECOOP '90 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990, pp. 140 - 150.
559. Maes, "Concepts and Experiments in Computational Reflection," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 147 - 155.
560. Malhotra, J.C. Thomas, J.M. Carroll, and L. Miller, "Cognitive Processes in Design," International Journal of Man-Machine Studies, Vol. 12, No. 2, February 1980, pp. 119 - 140.
561. Mandrioli and B. Meyer, Editors, Advances in Object-Oriented Software Engineering, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
562. Mannino, I.J. Choi, and D.S. Batory, "The Object-Oriented Functional Data Language," IEEE Transactions on Software Engineering, Vol. 16, No. 11, November 1990, pp. 1258 - 1272.
563. Marable and C.C. Belgrave, "Designing an Ada Tutorial Using Object-Oriented Design," Proceedings of the Eighth Annual National Conference on Ada Technology, March 1990, pp. 19 - 20.
564. Marca and C.L. McGowan, SADT -- Structured Analysis and Design Technique, McGraw-Hill Book Company, New York, New York, 1988.
565. Margono and E.V. Berard, "A Modified Booch's Taxonomy for Ada Generic Data-Structure Components and Their Implementation," Ada Components: Libraries and Tools -- Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Cambridge University Press, Cambridge, United Kingdom, 1987, pp. 61 - 74.
566. Margono and J.E. Walker, "An Object-Oriented Approach to Simulating a Real-Time System in Ada," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 239 - 244.
567. Martin and H. Fisher, "Integrating Ada Design Graphics Into the Ada Software Development Process," Proceedings of the Fifth Washington Ada Symposium, June 27 - 30, 1988, Association for Computing Machinery, New York, New York, 1988, pp. 163 - 170.

- 568. Martin and J.J. Odell, Object-Oriented Analysis and Design, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
- 569. Martin, Fourth Generation Languages, Volume I: Principles, Prentice Hall, Englewood Cliffs, New Jersey, 1985.
- 570. Martin, System Design from Provably Correct Constructs, Prentice Hall, Englewood Cliffs, New Jersey, 1985.
- 571. Martin, Information Engineering, Book 1: Introduction, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
- 572. Martin, Information Engineering, Book 2: Planning and Analysis, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
- 573. Martin, Information Engineering, Book 3: Design and Construction, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
- 574. Masiero and F.S.R. Germano, "JSD As An Object-Oriented Design Method," Software Engineering Notes, Vol. 13, No. 3, July 1988, pp. 22 - 23.
- 575. Masini, A. Napoli, D. Colnet, D. Leonard, and K. Tombre, Object Oriented Languages, Academic Press, New York, New York, 1991.
- 576. Masters and M.J. Kuchinski, "Software Design Prototyping Using Ada," Ada Letters, Vol. II, No. 4, January-February 1983, pp. 68 - 75.
- 577. McGraw and K. Harbison-Briggs, Knowledge Acquisition: Principles and Guidelines, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
- 578. McGregor and D.A. Sykes, Object-Oriented Software Development: Engineering Software for Reuse, Van Nostrand Reinhold, New York, New York, 1992.
- 579. McIntyre and L.F. Higgins, "Object-Oriented Systems Analysis and Design: Methodology and Application," Journal of Management Information Systems, Vol. 5, No. 1, Summer 1988. pp. 25 - 35.
- 580. McQuown, "Object-Oriented Design in a Real-Time Multiprocessor Environment," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 570 - 588.
- 581. Mellender, "An Integration of Logic and Object-Oriented Programming," SIGPLAN Notices, Vol. 23, No. 10, October 1988, pp. 181 - 185.
- 582. Merrow and J. Laursen, "A Pragmatic System for Shared Persistent Objects," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 103 - 110.
- 583. Meyer, M. Wallis, and M. Meier, "Experiences in Applying the Layered Virtual Machine/Object-Oriented Development Methodology to an Ada Design Effort," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment,

- October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 416 - 425.
- 584. Meyer, "Principles of Package Design," Communications of the ACM, Vol. 25, No. 7, July 1982, pp. 419 - 428.
 - 585. Meyer, "Genericity Versus Inheritance," OOPSLA Conference Proceedings, (Special Issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986), Association for Computing Machinery, New York, New York, 1986, pp. 391 - 405.
 - 586. Meyer, "Eiffel: Programming for Reusability and Extendability," SIGPLAN Notices, Vol. 22, No. 2, February 1987, pp. 85 - 94.
 - 587. Meyer, B., "Reusability: The Case for Object-Oriented Design," IEEE Software, Vol. 4, No. 2, March 1987, pp. 50-64.
 - 588. Meyer, Object-Oriented Software Construction, Prentice Hall, Englewood Cliffs, New Jersey, 1988.
 - 589. Meyer, "Tools for the New Culture: Lessons From the Design of the Eiffel Libraries," Communications of the ACM, Vol. 33, No. 9, September 1990, pp. 68 - 88.
 - 590. Meyer, Eiffel: The Language, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
 - 591. Meyer, "Applying `Design by Contract'," IEEE Computer, Vol. 25, No. 10, October 1992, pp. 40 - 51.
 - 592. Micallef, "Encapsulation, Reusability and Extensibility in Object-Oriented Programming Languages," Journal of Object-Oriented Programming, Vol. 1, No. 1, April/May 1988, pp. 12 - 36.
 - 593. Mickel, "Experiences With an Object-Oriented Method of Software Design," Proceedings of the Third Joint Ada Europe/AdaTech Conference, Cambridge University Press, Cambridge, United Kingdom, 1984, pp. 271 - 280.
 - 594. Miller, "The Magical Number Seven, Plus or Minus Two: Some Limits On Our Capacity for Processing Information," The Psychological Review, Vol. 63, No. 2, March 1956, pp. 81-97.
 - 595. Miller, "Multiple Inheritance In C++," Computer Language, Vol. 6, No. 8, August 1989, pp. 63 - 71.
 - 596. Millikin, "Object Orientation: What It Can Do For You," ComputerWorld, Vol. 23, No. 11. March 13, 1989, pp. 103 - 113.
 - 597. Milner, M. Tofte, and R. Harper, The Definition of Standard ML, MIT Press, Cambridge, Massachusetts, 1990.
 - 598. Milner, "A Theory of Type Polymorphism in Programming," Journal of Computer and System Sciences, Vol. 17, 1978, pp. 348 - 375.
 - 599. Milner, "A Proposal for Standard ML," ACM Symposium on Lisp and Functional Programming, August 1984, pp. 184 - 197.

600. Minsky, *The Society of Mind*, Simon and Schuster, New York, New York, 1985.
601. Mitchell, W. Maybury, and R. Sweet, *Mesa Language Manual (Version 5.0)*, Technical Report CSL-79-3, Xerox PARC, Palo Alto, California, April 1979.
602. Mohan and R.L. Kashyap, "An Object-Oriented Knowledge Representation for Spatial Information," *IEEE Transactions on Software Engineering*, Vol. 14, No. 5, May 1988, pp. 675 - 681.
603. Monarchi and G.I. Puhr, "A Research Typology for Object-Oriented Analysis and Design," *Communications of the ACM*, Vol. 35, No. 9, September 1992, pp. 35 - 47.
604. Monarchi, G. Booch, B. Henderson-Sellers, I. Jacobson, S. Mellor, J. Rumbaugh, and R. Wirfs-Brock, "Methodology Standards: Help or Hindrance?," *Proceedings of the Ninth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications*, *ACM SIGPLAN Notices*, Vol. 29, No. 10, October 1994, pp. 223 - 228.
605. Moon, "Object-Oriented Programming With Flavors," *OOPSLA Conference Proceedings*, (Special Issue of *SIGPLAN Notices*, Vol. 21, No. 11, November 1986), Association for Computing Machinery, New York, New York, 1986, pp. 1 - 8.
606. Morgan, "Configuration Management and Version Control in the Rational Programming Environment," in *Ada In Industry: Proceedings of the Ada-Europe International Conference Munich 7-9 June, 1988*, Cambridge University Press, Cambridge, United Kingdom, 1988, pp. 17 - 28.
607. Morrison, M.P. Atkinson, A.L. Brown, and A. Dearle, "Bindings in Persistent Programming Languages," *SIGPLAN Notices*, Vol. 23, No. 4, April 1988, pp. 27 - 34.
608. Moss, "Working With Persistent Objects: To Swizzle or Not to Swizzle," *IEEE Transactions on Software Engineering*, Vol. 18, No. 8, August 1992, pp. 657 - 673.
609. Moss and W.H. Kohler, "Concurrency Features for the Trellis/Owl Language," *Proceedings of the European Conference on Object-Oriented Programming 1987*, Paris, France, pp. 223 - 232.
610. Mrdalj, "Stepwise Object-Oriented System Design," *Proceedings of the 1990 IEEE International Conference on Computer Systems and Software Engineering -- EuroComp '90*, Tel-Aviv, Israel, May 8 - 10, 1990, pp. 520 - 521.
611. Mrdalj, "Bibliography of Object-Oriented System Development," *Software Engineering Notes*, Vol. 15, No. 5, October 1990, pp. 60 - 63.
612. Mullin, *Object-Oriented Program Design: With Examples in C++*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.
613. Mullin, *Rapid Prototyping for Object-Oriented Systems*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1990.
614. Myers, *Software Reliability: Principles and Practices*, John Wiley & Sons, New York, New York, 1976.
615. Myers, *Composite/Structured Design*, Van Nostrand Reinhold, New York, New York, 1978.

616. Myers, *Advances in Computer Architecture*, Second Edition, John Wiley & Sons, New York, New York, 1982.
617. Nahouraii and F. Petry, Editors, *IEEE Tutorial on Object-Oriented Databases*, IEEE Catalog Number EH0332-7, IEEE Computer Society Press, Los Alamitos, California, 1991.
618. Narotam and M. Caffey, "Object Oriented Development of Software Using Ada," *Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium*, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 354 - 359.
619. Naur and B. Randell, Editors, *Software Engineering: Report on a Conference Sponsored by the NATO Science Committee*, Garmisch, Germany, October 7-11, 1968.
620. Neighbors, "Software Construction Using Components," Technical Report 160, Department of Information and Computer Sciences, University of California, Irvine, 1980.
621. Neighbors, "The DRACO Approach to Constructing Software From Reusable Components," *IEEE Transactions on Software Engineering*, Vol. SE-10, No. 5, September 1984, pp. 564 - 574.
622. Nelson and G.F. Mota, "Object-Oriented Programming in Classic-Ada," *Ada Letters*, Vol. 12, No. 2, March/April 1992, pp. 102 - 110.
623. Nelson, G. Mota, and V. Theologitis, "Concurrent Object-Oriented Programming in Classic-Ada," *Ada Letters*, Vol. 12, No. 5, September/October 1992, pp. 77 - 83.
624. Nelson, "Concurrency and Object-Oriented Programming," *ACM SIGPLAN Notices*, Vol. 26, No. 10, October 1991, pp. 63 - 72.
625. -M. Nerson, "Applying Object-Oriented Analysis and Design," *Communications of the ACM*, Vol. 35, No. 9, September 1992, pp. 63 - 74.
626. Newton and J. Watkins, "The Combination of Logic and Objects for Knowledge Representation," *Journal of Object-Oriented Programming*, Vol. 1, No. 4, November/December 1988, pp. 7 - 10.
627. Nguyen and B. Hailpern, "A Generalized Object Model," *SIGPLAN Notices*, Vol. 21, No. 10, October 1986, pp. 78 - 87.
628. Nguyen and D. Rieu, *Schema Evolution in Object-Oriented Database Systems*, Technical Report No. 947, Unite De Recherche, INRIA-Rocquencourt, France, December 1988.
629. Nielsen, *Object-Oriented Design With Ada*, Bantam Books, New York, New York, 1992.
630. Nielsen and K. Shumate, "Designing Large Real-Time Systems With Ada," *Communications of the ACM*, Vol. 30, No. 8, August 1987, pp. 695 - 715.
631. Niestrasz, "A Survey of Object-Oriented Concepts," in *Object-Oriented Concepts, Databases, and Applications*, ACM Press/Addison Wesley, Reading, Massachusetts, 1989, pp. 3 - 21.

632. Nierstrasz, S. Gibbs, and D. Tsichritzis, "Component-Oriented Software Development," Communications of the ACM, Vol. 35, No. 9, September 1992, pp. 160 - 165.
633. Nissen and P. Wallis, Portability and Style In Ada, Cambridge University Press, Cambridge, United Kingdom, 1984.
634. Northrop, "Object-Oriented Development," The Encyclopedia of Software Engineering, Volume 2, J.J. Marciniak, Editor, John Wiley and Sons, New York, New York, 1994, pp. 729 - 737.
635. Novobilski, "Pictorial Design Notation for Software-ICs," Journal of Object-Oriented Programming, Vol. 3, No. 2, July/August 1990, pp. 9 - 14.
636. Nygaard, "Basic Concepts in Object Oriented Programming," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 128 - 132.
637. O'Brien, D.C. Halbert, and M.F. Kilian, "The Trellis Programming Environment," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 91 - 102.
638. Olan, "Unconventional Design," Computer Language, Vol. 5, No. 5, May 1988, pp. 36 - 41.
639. Organick, A Programmer's View of the Intel 432 System, McGraw-Hill, New York, New York, 1983.
640. Page-Jones, The Practical Guide to Structured Systems Design, Second Edition, Yourdon Press/Prentice Hall, Englewood Cliffs, New Jersey, 1988.
641. Page-Jones, "Comparing Techniques by Means of Encapsulation and Connascence," Communications of the ACM, Vol. 35, No. 9, September 1992, pp. 147 - 151.
642. -S. Park, "Abstract Object Types = Abstract Knowledge Types + Abstract Data Types + Abstract Connector Types," Journal of Object-Oriented Programming, Vol. 4, No. 3, June 1991, pp. 37 - 39, 42 - 44, 46 - 48, 51 - 52.
643. Parnas, "On the Criteria To Be Used in Decomposing Systems Into Modules," Communications of the ACM, Vol. 5, No. 12, December 1972, pp. 1053-1058.
644. Partridge, Artificial Intelligence: Applications in the Future of Software Engineering, Halstead Press, New York, New York, 1986.
645. Pasquier-Boltuck, E. Grossman, and G. Collaud, "Prototyping An Interactive Electronic Book System Using an Object-Oriented Approach," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 177 - 190.
646. Premerlani, M.R. Blaha, J.E. Rumbaugh, and T.A. Varwig, "An Object-Oriented Relational DBMS," Communicaitons of the ACM, Vol. 33, No. 11, November 1990, pp. 99 - 109.
647. Pendley, "Using Information Engineering and Ada Object-Oriented Design Methods in Concert -- A Case Study," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 11 - 19.

648. Perez, "Simulating Inheritance With Ada," Ada Letters, Vol. 8, No. 5, September-October, 1988, pp. 37 - 46.
649. Perry and G.E. Kaiser, "Adequate Testing and Object-Oriented Programming," Journal of Object-Oriented Programming, Vol. 2, No. 5, January/February 1990, pp. 13 - 19.
650. Peterson, "Petri Nets," ACM Computing Surveys, Vol. 9, No. 3, September 1977, pp. 223 - 252.
651. Peterson, Petri Net Theory and the Modeling of Systems, Prentice Hall, Englewood Cliffs, New Jersey, 1981.
652. Peterson, Tutorial: Object-Oriented Computing, Volume 1: Concepts, IEEE Catalog Number EH0257-6, IEEE Computer Society Press, Washington, D.C., 1987.
653. Peterson, Tutorial: Object-Oriented Computing, Volume 2: Implementations, IEEE Catalog Number EH0257-6, IEEE Computer Society Press, Washington, D.C., 1987.
654. Premerlani, M.R. Blaha, J.E. Rumbaugh, and T.A. Varwig, "An Object-Oriented Relational DBMS," Communications of the ACM, Vol. 33, No. 11, November 1990, pp. 99 - 109.
655. Pendley, "Using Information Engineering and Ada Object-Oriented Design Methods in Concert -- A Case Study," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 11 - 19.
656. Perez, "Simulating Inheritance With Ada," Ada Letters, Vol. 8, No. 5, September-October, 1988, pp. 37 - 46.
657. Perry and G.E. Kaiser, "Adequate Testing and Object-Oriented Programming," Journal of Object-Oriented Programming, Vol. 2, No. 5, January/February 1990, pp. 13 - 19.
658. Peterson, "Petri Nets," ACM Computing Surveys, Vol. 9, No. 3, September 1977, pp. 223 - 252.
659. Peterson, Petri Net Theory and the Modeling of Systems, Prentice Hall, Englewood Cliffs, New Jersey, 1981.
660. Peterson, Tutorial: Object-Oriented Computing, Volume 1: Concepts, IEEE Catalog Number EH0257-6, IEEE Computer Society Press, Washington, D.C., 1987.
661. Peterson, Tutorial: Object-Oriented Computing, Volume 2: Implementations, IEEE Catalog Number EH0257-6, IEEE Computer Society Press, Washington, D.C., 1987.
662. Pun and R.L. Winder, "A Design Method for Object-Oriented Programming," ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 225 - 240.
663. Purdy, B. Schuchardt, and D. Maier, "Integrating an Object Server With Other Worlds," ACM Transactions on Office Information Systems, Vol. 5, No. 1, January 1987, pp. 27 - 47.

- 664. Rajlich and J. Silva, "Two Object-Oriented Decomposition Techniques," Proceedings of the Fifth Washington Ada Symposium, June 27 - 30, 1988, Association for Computing Machinery, New York, New York, 1988, pp. 171 - 176.
- 665. Rajlich, "Paradigms for Design and Implementation in Ada," Communications of the ACM, Vol. 28, No. 7, July 1985, pp. 718 - 727.
- 666. Ramamoorthy, V. Garg and A. Prakash, "Programming in the Large," IEEE Transactions on Software Engineering, Vol. SE-12, No. 7, July 1986, pp. 769 - 783.
- 667. Rapin, "Procedural Objects In Newton," SIGPLAN Notices, Vol. 24, No. 9, September 1989, pp. 133 - 135.
- 668. Ratcliffe, "Report on a Workshop on Software Reuse Held at Hereford, UK on 1,2 May 1986," Software Engineering Notes, Vol. 12, No. 1, January 1987, pp. 42 - 47.
- 669. Rational, Inc., Large-System Development and Rational Subsystems, Document Control Number 6004, Rational, Inc., Mountain View, California, November, 1986.
- 670. Rattray, Editor, Specification and Verification of Concurrent Systems, Springer-Verlag, London, United Kingdom, 1990.
- 671. Reed and Donald E. Bynum, "Analyzing Systems for Object-Oriented Design," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 195 - 200.
- 672. Reichbach and R.A. Kemmerer, "Soundworks: An Object-Oriented Distributed System for Digital Sound," IEEE Computer, Vol. 25, No. 3, March 1992, pp. 25 - 37.
- 673. Reid, "Object-Oriented Requirements Analysis: A Tool Vision," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 227 - 230.
- 674. Reisig, Petri Nets: An Introduction, Springer-Verlag, New York, New York, 1985.
- 675. Reiss, "An Object-Oriented Framework for Graphical Programming," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 49 - 57.
- 676. Reiss, "Working in the Garden Environment for Conceptual Programming," IEEE Software, Vol. 4, No. 6, November 1987, pp. 16 - 27.
- 677. Rentsch, "Object Oriented Programming," SIGPLAN Notices, Vol. 17, No. 9, September 1982, pp. 51 - 57.
- 678. Rine, "A Brief Comparison of Ada and Object-Oriented Design Elements for Ada," Proceedings of the Second Annual Conference on Artificial Intelligence and Ada, 1986, pp. 10-1 - 10-10.
- 679. Rine, "A Common Error in the Object Structure of Object-Oriented Design Methods," Software Engineering Notes, Vol. 12, No. 4, October 1987, pp. 42 - 44.
- 680. Rine, "A Proposed Standard Set of Principles for Object-Oriented Development," Software Engineering Notes, Vol. 16, No. 1, January 1991, pp. 43 - 49.

681. Rocky Mountain Institute of Software Engineering, Workshop on Software Reuse -- Participant Proceedings, October 1987, summary contained in P. Robinson, Editor, Object-Oriented Design, Chapman Hall, London, United Kingdom, 1992.
682. Robinson, Hierarchical Object-Oriented Design, Prentice Hall, Englewood Cliffs, New Jersey, 1992.
683. Robinson and G. Berrisford, Object-Oriented SSADM, Prentice Hall, Englewood Cliffs, New Jersey, 1994.
684. Robinson and K. Leavitt, "Proof Techniques for Hierarchically Structured Programs," in Current Trends in Programming Methodologies, Volume 2, Raymond T. Yeh, Editor, Prentice Hall, Englewood Cliffs, New Jersey, 1977.
685. Rosenstein, K. Doyle, and S. Wallace, "Object Oriented Programming for Macintosh Applications," Proceedings of the 1986 Fall Joint Computer Conference, IEEE Catalog Number 86CH2345-7, IEEE Computer Society Press, Washington, D.C., 1986, pp 31 - 35.
686. Ross and K.E. Schoman, "Structured Analysis for Requirements Definition," IEEE Transactions on Software Engineering, Vol. SE-3, No. 1, January 1977, pp. 69 - 84.
687. Ross, J.B. Goodenough, and C.A. Irvine, "Software Engineering: Process, Principles, and Goals," IEEE Computer, Vol. 8, No. 5, May 1975, pp. 17 - 27.
688. Ross, "The Form of a Passive Iterator," Ada Letters, Vol. 9, No. 2, March/April 1989, pp. 102 - 105.
689. Ross, "Issues in Object-Oriented Requirements Analysis," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 77 - 99.
690. Rosson and E. Gold, "Problem-Solution Mapping In Object-Oriented Design," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 7 - 10.
691. Royce, "Managing the Development of Large Software Systems: Concepts and Techniques," Proceedings of WESCON, August 1970.
692. Rubin and A. Goldberg, "Object Behavior Analysis," Communications of the ACM, Vol. 35, No. 9, September 1992, pp. 48 - 62.
693. Rumbaugh, "Relations As Semantic Constructs in an Object-Oriented Language," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 466 - 481.
694. Rumbaugh, "Controlling Propagation of Operations Using Attributes on Relations," OOPSLA '88 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 285 - 296.
695. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen, Object-Oriented Modeling and Design, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
696. Russell, "Experiences Implementing a Reusable Data Structure Component Taxonomy," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and

- Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 8 - 18.
697. Safford, "Ada Object-Oriented Design Saves Costs," Government Computer News, Vol. 6, No. 19, September 25, 1987. page 108.
 698. Sakkinen, "Comments on 'the Law of Demeter' and C++," SIGPLAN Notices, Vol. 23, No. 12., December 1988, pp. 38 - 44.
 699. Sakkinen, "Disciplined Inheritance," ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 39 - 56.
 700. Savage, "Sun and HP Pledge Oneness In Object-Oriented Business," ComputerWorld, Vol. XXV, No. 9, March 4, 1991, page 40.
 701. Scannell, "Freeform DBMS the 'Object' of Startup Company's Affection," Mini-Micro Systems, Vol. XXI, No. 2, February 1988, pp. 16 - 22.
 702. Schaffert, T. Cooper, B. Bullis, M. Killian, and C. Wilpolt, "An Introduction to Trellis/Owl," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 9 - 16.
 703. Schalkoff, Artificial Intelligence: An Engineering Approach, McGraw-Hill Publishing Company, New York, New York, 1990.
 704. R Schank and P. Childers, The Creative Attitude: Learning to Ask and Answer the Right Questions, Macmillan Publishing Company, New York, New York, 1988.
 705. Scharenberg and H.E. Dunsmore, "Evolution of Classes and Objects During Object-Oriented Design and Programming," Journal of Object-Oriented Programming, Vol. 3, No. 5, January/February 1991, pp. 18 - 28.
 706. Schaschinger, "ESA, An Expert Supported OOA Method and Tool," Software Engineering Notes, Vol. 17, No. 2, April 1992, pp. 50 - 56
 707. Scheidt, D. Preston, and M. Armstrong, "Implementing Semantic Networks in Ada," Proceedings of the Second Annual Conference on Artificial Intelligence and Ada, 1986, pp. 8-1 - 8-16.
 708. Schmidt, Denotational Semantics -- A Methodology For Language Development, Allyn and Bacon, Inc., Newton, Massachusetts, 1986.
 709. Schmucker, Object-Oriented Programming for the Macintosh, Hayden, Hasbrouck Heights, New Jersey, 1986.
 710. Schmucker, "Object Orientation," MacWorld, Vol. 3, No. 11, November 1986, pp. 119 - 123.
 711. Schoch, "An Overview of the Language Smalltalk-72," SIGPLAN Notices, Vol. 14, No. 9, September 1979, pp. 64 - 73.

712. Schrefl and E.J. Neuhold, "Object Class Definition By Generalization Using Upward Inheritance," Proceedings of the Fourth International Conference on Data Engineering, 1988, pp. 4 - 13.
713. Sedbrook, "Improvisational Exercises for Hatching Objects," Journal of Object-Oriented Programming, Vol. 3, No. 4, November/December 1990, pp. 40 - 42.
714. Seidewitz and M. Stark, "Towards a General Object-Oriented Software Development Methodology," Proceedings of the First International Conference on Ada Programming Language Applications for the NASA Space Station, Vol. II, Edited by R.L. Bown, University of Houston - Clear Lake, June 1986, pp. D.4.6.1 - D.4.6.14.
715. Seidewitz and M. Stark, General Object-Oriented Software Development, Document No. SEL-86-002, NASA Goddard Space Flight Center, Greenbelt, Maryland, 1986.
716. Seidewitz, "Object-Oriented Programming In Smalltalk and Ada," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 202 - 213.
717. Seidewitz, "Object-Oriented Programming Through Type Extension in Ada 9X," Ada Letters, Vol. 11, No. 2, March/April 1991, pp. 86 - 97.
718. Seidewitz, "Object-Oriented Programming With Mixins in Ada," Ada Letters, Vol. 12, No. 2, March/April 1992, pp. 76 - 90.
719. Selic, G. Gullekson, J. McGee, and I. Engelberg, "ROOM: An Object-Oriented Methodology for Developing Real-Time Systems," Proceedings of the 5th International Workshop on CASE, Montreal, Canada, July 1992.
720. Selic, G. Gullekson, and P.T. Ward, Real-Time Object-Oriented Modelling, John Wiley and Sons, New York, New York, 1994.
721. Shah, J.E. Rumbaugh, J.H. Hamel, and R.A. Borsari, "DSM: An Object-Relationship Modeling Language," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 191 - 202.
722. Shankar, "A Functional Approach to Module Verification," IEEE Transactions on Software Engineering, Vol. SE-8, No. 2, March 1982, pp. 147 - 160.
723. Shapiro, Y. Gourhant, S. Habert, L. Mosseri, M. Ruffin, and C. Valot, "SOS: An Object-Oriented Operating System -- Assessment and Perspectives," Computing Systems, Vol. 2, No. 4, December 1989, pp. 287 - 338.
724. Shaw, W.A. Wolf, and R. London, "Abstraction and Verification in Alphard: Iteration and Generators," Alphard: Form and Content, Springer-Verlag, New York, New York, 1981, pp. 73 - 116.
725. Shaw, Editor, Alphard: Form and Content, Springer-Verlag, New York, New York, 1981.
726. Shaw, "Abstraction Techniques in Modern Programming Languages," IEEE Software, Vol. 1, No. 4, October 1984, pp. 10 - 26.
727. Shlaer and S.J. Mellor, Object-Oriented Systems Analysis: Modeling the World In Data, Yourdon Press: Prentice Hall, Englewood Cliffs, New Jersey, 1988.

728. Shlaer and S.J. Mellor, "An Object-Oriented Approach to Domain Analysis," Software Engineering Notes, Vol. 14, No. 5, July 1989, pp. 66 - 77.
729. Shlaer and S.J. Mellor, Object Lifecycles: Modeling the World In States, Yourdon Press: Prentice Hall, Englewood Cliffs, New Jersey, 1992.
730. Shopiro, "An Example of Multiple Inheritance in C++: A Model of the Iostream Library," SIGPLAN Notices, Vol. 24, No. 12, December 1989, pp. 32 - 36.
731. Shriver and P. Wegner, Editors, Research Directions in Object-Oriented Programming, The MIT Press, Cambridge, Massachusetts, 1987.
732. Shumate, "Layered Virtual Machine/Object-Oriented Design," Proceedings of the Fifth Washington Ada Symposium, June 27 - 30, 1988, Association for Computing Machinery, New York, New York, 1988, pp. 177 - 190.
733. Simon, The Sciences of the Artificial, Second Edition, MIT Press, Cambridge, Massachusetts, 1981.
734. Simonian and M. Crone, "INNOVADA: An Object-Oriented Ada Environment," Proceedings of the U.S. Army Information Systems Engineering Command Advanced Technology Office -- Technology Strategies '88 Conference, February 9-12, 1988, The American Defense Preparedness Association, Washington, D.C., pp. 63 - 74.
735. Simonian and M. Crone, "InnovAda: True Object-Oriented Programming In Ada," Journal of Object-Oriented Programming, Vol. 1, No. 4, November/December 1988, pp. 14 - 21.
736. Simos, "The Domain-Oriented Software Life-Cycle: Towards and Extended Process Model for Reusability," from L. Singh, "Interview With Bjarne Stroustrup," The C++ Journal, Vol. 1, No. 3, 1991, pp. 16 - 18, 20 - 25.
737. Sively, "Experience and Lessons Learned in Transporting Ada Software," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 436 - 440.
738. Sively, "Ada Software Productivity Issues: Results From the AFATDS Project," Proceedings of the TRI-Ada '88 Conference, October 24-27, 1988, ACM Press, New York, New York, 1988, pp. 543 - 555.
739. Sixtensson and W. Ye, "Reuse in the Telecommunications Domain Using Object-Oriented Technology and Ada," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 231 - 239.
740. Smith and S.R. Tockey, "An Integrated Approach to Software Requirements Definition Using Objects," Proceedings of Ada Expo 1988, Galaxy Productions, Frederick, Maryland, 1988, 21 pages.
741. Smith, P.S. Barth, and R.L. Young, "A Substrate for Object-Oriented Interface Design," in Research Directions in Object-Oriented Programming, B. Shriver and P. Wegner, Editors, The MIT Press, Cambridge, Massachusetts, 1987, pp. 253 - 315.

742. Smith, Concepts of Object-Oriented Programming, McGraw-Hill, New York, New York, 1991.
743. Snyder, "CommonObjects: An Overview," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 19 - 28.
744. Snyder, "Encapsulation and Inheritance in Object-Oriented Programming Languages," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 38 - 45.
745. Snyder, "Inheritance and the Development of Encapsulated Software Components," Proceedings of the Twentieth Hawaii International Conference on System Sciences, Kona, Hawaii, January 1987, pp. 227 - 238.
746. Snyder, "Inheritance and the Development of Encapsulated Software Components," Research Directions in Object-Oriented Programming, The MIT Press, Cambridge, Massachusetts, 1987, pp. 165 - 188.
747. Sowa, Conceptual Structures: Information Processing in Mind and Machine, Addison-Wesley, Reading, Massachusetts, 1984.
748. Spicer and D.A. Umphress, "A Method for Mapping an Analysis to a Reusable Design," Ada Letters, Vol. XI, No. 9, November/December 1991, pp. 67 - 82.
749. St. Dennis, P. Stachour, E. Frankowski, and E. Onuegbu, "Measurable Characteristics of Reusable Ada Software," Ada Letters, Vol. VI, No. 2, March-April 1986, pp. 41 - 50.
750. Stadel, "Object-Oriented Programming Techniques to Replace Software Components on the Fly," SIGPLAN Notices, Vol. 26, No. 1, January 1991, pp. 99 - 108.
751. Stark and E.V. Seidewitz, "Towards a General Object-Oriented Ada Life-Cycle," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 213 - 222.
752. Stefik and D.G. Bobrow, "Object-Oriented Programming: Themes and Variations," The AI Magazine, Vol. 6, No. 4, Winter 1986, pp. 40 - 62.
753. Steigerwald and M.L. Nelson, "Concurrent Programming in Smalltalk-80," SIGPLAN Notices, Vol. 25, No. 8, August 1990, pp. 27 - 36.
754. Stein, "Delegation Is Inheritance," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 138 - 146.
755. Stepney, R. Barden, and D. Cooper, Editors, Object-Orientation in Z, Springer-Verlag, London, United Kingdom, 1992.
756. Stevens, G.J. Myers and L.L. Constantine, "Structured Design," IBM Systems Journal, Vol. 13, No. 2, May 1974, pp. 115 - 139.
757. Stoecklin, E.J. Adams, and S. Smith, "Object-Oriented Analysis," Proceedings of the Fifth Washington Ada Symposium, June 27 - 30, 1988, Association for Computing Machinery, New York, New York, 1988, pp. 133 - 138.

758. Stone and D. Hentchel, "Database Wars Revisited," Byte, Vol. 15, No. 10, October 1990, pp. 233 - 234, 236, 238, 240, 242.
759. Stoy, Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory, MIT Press, Cambridge, Massachusetts, 1977.
760. Strachey, Fundamental Concepts in Programming Languages, Lecture Notes, International Summer School in Computer Programming, Copenhagen, August 1967.
761. Strom and S. Yemini, "NIL: An Integrated Language and System for Distributed Programming," SIGPLAN Notices, Vol. 18, No. 6, June 1983, pp. 73 - 82.
762. Strom, "A Comparison of the Object-Oriented and Process Paradigms," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 88 - 89.
763. Stroustrup, "Data Abstraction in C," AT&T Bell Laboratories Technical Journal, Vol. 63, No. 8, Part 2, October 1984, pp. 1701 - 1732.
764. Stroustrup, The C++ Programming Language, Addison-Wesley, Reading, Massachusetts, 1986.
765. Stroustrup, "An Overview of C++," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 7 - 18.
766. Stroustrup, "What Is Object-Oriented Programming," IEEE Software, Vol. 5, No. 3, May 1988, pp. 10 - 20.
767. Stroustrup, "Parameterized Types for C++," Proceedings of the C++ Conference, Denver, Colorado, October 1988, USENIX Association, Berkeley, California, 1988, pp. 1 - 18.
768. Stroustrup, The C++ Programming Language, Second Edition, Addison-Wesley, Reading, Massachusetts, 1991.
769. Sutherland, Sketchpad, A Man-Machine Graphical Communication System, Ph. D. Thesis, Massachusetts Institute of Technology, January 1963.
770. Suzuki, "Inferring Types In Smalltalk," Conference Record of the Eighth Annual ACM Symposium on Principles of Programming Languages, 1981, pp. 187 - 199.
771. Taenzer, M. Ganti, and S. Podar, "Problems in Object-Oriented Software Reuse," ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 25 - 38.
772. Tang, "C++'s Destructors Can Be Destructive," ACM SIGPLAN Notices, Vol. 26, No. 10, October 1991, pp. 44 - 52.
773. Tarumi, K. Agusa, and Y. Ohno, "A Programming Environment Supporting Reuse of Object-Oriented Software," Proceedings of the 10th International Conference on Software Engineering, April 11-15, 1988, pp. 265 - 273.
774. Tasker, "Object Lesson," ComputerWorld, Vol. XXV, No. 16, April 22, 1991, pp. 79 - 81, 84.

775. Taylor and A. Hecht, "Using CASE for Object-Oriented Design with C++," Computer Language, Vol. 7, No. 11, November 1990, pp. 49 - 57.
776. Taylor, Object-Oriented Technology: A Manager's Guide, Addison-Wesley, Reading, Massachusetts 1990.
777. Taylor, Business Engineering With Object Technology, Addison-Wesley, Reading, Massachusetts, 1995.
778. Tello, Object-Oriented Programming for Artificial Intelligence, Addison-Wesley, Reading, Massachusetts, 1989.
779. Temte, "Let's Begin Introducing the Object-Oriented Paradigm," SIGCSE Bulletin, Vol. 23, No. 1, March 1991, pp. 73 - 77.
780. Ten Dyke and J.C. Kunz, "Object-Oriented Programming," IBM Systems Journal, Vol. 28, No. 3, 1989, pp. 465 - 478.
781. Tesler, "Object Pascal Report," Structured Language World, Vol. 9, No. 3, 1985, pp. 10 - 14.
782. Thomas and K. Johnson, "Orwell: A Configuration Management System for Team Programming," SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 135 - 141.
783. Thomas, "What's An Object?," Byte, Vol. 14, No. 3, March 1989, 231 - 244.
784. Thomas, "In Search of an Object-Oriented Development Process," Journal of Object-Oriented Programming, Vol. 2, No. 1, May/June 1989, pp. 60 - 63.
785. Toetenel, J. van Katwijk, and N. Plat, "Structured Analysis -- Formal Design, Using Stream and Object-Oriented Formal Specification," Proceedings of the ACM SIGSOFT International Workshop on Formal Methods in Software Development, Special Issue of Software Engineering Notes, Vol. 15, No. 4, September 1990, pp. 118 - 127.
786. Tomlinson and V. Singh, "Inheritance and Synchronization with Enabled-Sets," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 103 - 112.
787. Touati, "Is Ada an Object-Oriented Programming Language?," SIGPLAN Notices, Vol. 22, No. 5, May 1987, pp. 23 - 26.
788. Tracz, "Ada Reusability Efforts: A Survey of the State of the Practice," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 35 - 44.
789. Tracz, Editor Software Reuse: Emerging Technology, IEEE Catalog Number EH0278-2, Computer Society Order Number 846, IEEE Computer Society Press, Washington, D.C., 1988.
790. Trausan-Matu, J. Tepandi, and M. Barbuseanu, "Validation, Verification, and Testing of Object-Oriented Programs," Proceedings of the EastEurOOPe '91 Conference, September 15-19, 1991, Bratislava, Czechoslovakia, pp. 62 - 71.

791. Tripathi and M. Aksit, "Communication, Scheduling, and Resource Management in SINA," *Journal of Object-Oriented Programming*, Vol. 1, No. 4, November/December 1988, pp. 24 - 36.
792. Tripathi, A. Ghonami, and T. Schmitz, "Object Management in the NEXUS Distributed Operating System," *Digest of Papers COMPCON*, Spring 1987, IEEE Catalog Number 87CH2409-1, Computer Society Order Number 764, Computer Society Press of the IEEE, Washington, D.C., pp. 50 - 53.
793. A Tripathi, E. Berge, and M. Aksit, "An Implementation of the Object-Oriented Concurrent Programming Language SINA," *Software Practice and Experience*, Vol. 19, No. 3, March 1989, pp. 235 - 256.
794. Tsichritzis, E. Fiume, S. Gibbs and O.M. Nierstrasz, "KNOs: KNowledge Acquisition, Dissemination and Manipulation Objects," *ACM Transactions on Office Information Systems*, Vol. 5, No. 1, January 1987, pp. 96 - 112.
795. Turner, "Understanding the Elements of System Design," in *Critical Issues in Information Systems Research*, Edited by R.J. Boland, Jr. and R.A. Hirschheim, John Wiley & Sons, Chichester, U.K., 1987, pp. 97 - 111.
796. Ullman, *Principles of Database Systems*, 2nd Ed., Computer Science Press, Rockville, Maryland, 1982.
797. Ungar and D. Patterson, "What Price Smalltalk," *IEEE Computer*, Vol. 20, No. 1, January 1987, pp. 67 - 74.
798. Ungar and R.B. Smith, "Self: The Power of Simplicity," *OOPSLA '87 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 22, No. 12, December 1987, pp. 227 - 242.
799. Ungar, R.B. Smith, C. Chambers, and U. Hölzle, "Object, Message, and Performance: How They Coexist in Self," *IEEE Computer*, Vol. 25, No. 10, October 1992, pp. 53 - 64.
800. Ungar, *The Design and Evaluation of a High Performance Smalltalk System*, MIT Press, Cambridge, Massachusetts, 1987.
801. Urlocker, "A Methodology for the Real World," *Computer Language*, Vol. 6, No. 1, January 1989, pp. 47 - 58.
802. Urlocker, "Teaching Object-Oriented Programming," *Journal of Object-Oriented Programming*, Vol. 2, No. 2, July/August 1989, pp. 45 - 47.
803. *USENIX Association, Proceedings and Additional Papers of the C++ Workshop*, Santa Fe, New Mexico, November 1987, USENIX Association, Berkeley, California, 1987.
804. *USENIX Association, Proceedings of the C++ Conference*, Denver, Colorado, October 1988, USENIX Association, Berkeley, California, 1988.
805. *USENIX Association, Proceedings of the C++ Conference*, San Francisco, California, April 1990, USENIX Association, Berkeley, California, 1990.
806. Vick and C.V. Ramamoorthy, Editors, *Handbook of Software Engineering*, Van Nostrand Reinhold, New York, New York, 1984.

807. Vidale and C.R. Hayden, "A Student Project to Extend Object-Oriented Design," Proceedings of the Ada Software Engineering Education and Training Symposium, June 9-11, 1987, pp. 89 - 98.
808. Vielcanet, "HOOD Design Method and Control/Command Techniques for the Development of Realtime Software," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 213 - 219.
809. Vlissides and M.A. Linton, "Applying Object-Oriented Design to Structured Graphics," Proceedings of the C++ Conference, Denver, Colorado, October 1988, USENIX Association, Berkeley, California, 1988, pp. 81 - 94.
810. von Simpson, "Programming's New Object-Ive," ComputerWorld, Vol. XXIV, No. 22, May 28, 1990, page 43.
811. Vose, "Review of the C++ Programming Language," Byte, Vol. 11, No. 8, August 1986, pp. 63 - 68.
812. Wadler, "How to Make Ad Hoc Polymorphism Less Ad Hoc," Proceedings of the ACM Symposium on Principles of Programming Languages, 1989, pp. 60 - 76.
813. Walden and J.M. Nerson, Seamless Object-Oriented Software Architecture: Analysis and Design of Reliable Systems, Prentice Hall, Englewood Cliffs, New Jersey, 1995.
814. Walker, "Requirements Of An Object-Oriented Design Method,' Software Engineering Journal, Vol. 7, No. 2, March 1992, pp. 102 - 113.
815. Walters, "An Ada Object-Based Analysis and Design Approach," Ada Letters, Vol. XI, No. 5, July/August 1991, pp. 62 - 78.
816. Wand and R. Weber, "An Ontological Model of an Information System," IEEE Transactions on Software Engineering, Vol. 16, No. 11, November 1990, pp. 1282 - 1292.
817. Wand, "Complete Type Inference for Simple Objects," Proceedings of the IEEE Symposium on Logic in Computer Science, 1987, pp. 37 - 44.
818. Wand, "Type Inference for Record Concatenation and Multiple Inheritance," Proceedings of the IEEE Symposium on Logic in Computer Science, 1989, pp. 92 - 97.
819. Ward and S.J. Mellor, Structured Development for Real-Time Systems, Volumes 1-3, Yourdon Press, New York, New York, 1985.
820. Ward, "How to Integrate Object Orientation with Structured Analysis and Design," IEEE Software, Vol. 6, No. 2, March 1989, pp. 74 - 82.
821. Wartik, "File Structures for Object-Oriented Software Development," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 411 - 419.
822. Wasserman, P. Pircher, and R.J. Muller, "An Object-Oriented Design Method for Code Generation," Software Engineering Notes, Vol. 14, No. 1, January 1989, pp. 32 - 55.

- 823. Wasserman, P. Pircher, and R.J. Muller, "An Object-Oriented Design Notation for Software Design Representation," IEEE Computer, Vol. 23, No. 3, March 1990, pp. 50 - 63.
- 824. Watanabe and A. Yonezawa, "Reflection in an Object-Oriented Concurrent Language," OOPSLA '88 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 23, No. 11, November 1988, pp. 306 - 315.
- 825. Watanabe and A. Yonezawa, "Reflective Computation in Object-Oriented Concurrent Systems and Its Applications," Proceedings of the Fifth International Workshop On Software Specification and Design, May 19-20, 1989, Pittsburgh, Pennsylvania, IEEE Computer Society Press, Washington, D.C., May 1989, pp. 56 - 59.
- 826. Waurzyniak, "Borland Pulls Object Orientation Into Its Fold," ComputerWorld, Vol. 23, No. 18, May 1, 1989, page 37.
- 827. Webster, Pitfalls of Object-Oriented Development, M&T Books, New York, New York, 1995.
- 828. Wegmann, "Object-Oriented Programming Using Modula-2," Journal of Pascal, Ada & Modula-2, Vol. 5, No. 3, May-June 1986, pp. 5 - 17.
- 829. Wegner and S.B. Zdonik, "Inheritance as an Incremental Modification Mechanism, or What Like Is and Isn't Like," ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming, Lecture Note on Computer Science, Volume 322, Springer Verlag, New York, New York, 1988, pp. 55 - 77.
- 830. Wegner, "On the Unification of Data and Program Abstraction in Ada," Proceedings of the 10th Annual ACM Symposium on Principles of Programming Languages, 1983, pp. 256 - 264.
- 831. Wegner, "Classification in Object-Oriented Systems," SIGPLAN Notices, Vol. 21, No. 10, October 1986, pp. 173 - 182.
- 832. Wegner, "The Object-Oriented Classification Paradigm," in Research Directions in Object-Oriented Programming, B. Shriver and P. Wegner, Editors, The MIT Press, Cambridge, Massachusetts, 1987, pp. 479 - 560.
- 833. Wegner, "Dimensions of Object-Based Language Design," OOPSLA '87 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 22, No. 12, December 1987, pp. 168 - 182.
- 834. Wegner, "Workshop on Object-Oriented Programming ECOOP, Paris, June 18, 1987," SIGPLAN Notices, Vol. 23, No. 1, pp. 16 - 37.
- 835. Wegner, "Learning the Language," Byte, Vol. 14, No. 3, March 1989, pp. 245 - 253.
- 836. Wegner, "Concepts and Paradigms of Object-Oriented Programming," OOPS Messenger, Vol. 1, No. 1, August 1990, pp. 7 - 87.
- 837. Wegner, "Dimensions of Object-Oriented Modeling," IEEE Computer, Vol. 25, No. 10, October 1992, pp. 12 - 20.
- 838. Weinberg, The Psychology of Computer Programming, Van Nostrand Reinhold Company, New York, New York, 1971.

839. Weiskamp and B. Flamig, *The Complete C++ Primer*, Academic Press, Inc. (Harcourt Brace Jovanovich, Publishers) Boston, Massachusetts, 1990.
840. Weiss, "Why Object-Oriented Databases?," *Electronic Engineering Times*, No. 465, December 21, 1987, page 23.
841. Wells, J.A. Blakeley, and C.W. Thompson, "Architecture of an Open Object-Oriented Database Management System," *IEEE Computer*, Vol. 25, No. 10, October 1992, pp. 74 - 82.
842. Weston, *Elements of C++ Macintosh Programming*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1990.
843. Wets and M. Leon, "Ada For CATCAS(2): A New Era In Air Traffic Control," in *Ada In Industry: Proceedings of the Ada-Europe International Conference Munich 7-9 June, 1988*, Cambridge University Press, Cambridge, United Kingdom, 1988, pp. 94 - 103.
844. Wexelblat, Editor, *History of Programming Languages*, Academic Press, New York, New York, 1981.
845. Wheeler, "Object Data Bases and Ada," *Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium*, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 360 - 366.
846. Whitcomb and B. Clark, "Pragmatic Definition of an Object-Oriented Development Process for Ada," *Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment*, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 380 - 399.
847. White, *Rational ROSE Essentials: Using the Booch Method*, Benjamin/Cummings, Menlo Park, California, 1994.
848. Whitmire, "Object-Oriented Measurement of Software," *The Encyclopedia of Software Engineering*, Volume 2, J.J. Marciniak, Editor, John Wiley and Sons, New York, New York, 1994, pp. 737 - 739.
849. Whitting, "The Quest for a Better Way to Develop Software," *Electronic Business*, July 10, 1989, pp. 16 - 17.
850. Wiebe, "A Distributed Repository for Immutable Persistent Objects," *OOPSLA '86 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 21, No. 11, November 1986, pp. 453 - 465.
851. Wiederhold, "Views, Objects, and Databases," *IEEE Computer*, Vol. 19, No. 12, December 1986, pp. 37 - 44.
852. Wiener and L.J. Pinson, "A Practical Example of Multiple Inheritance in C++," *SIGPLAN Notices*, Vol. 24, No. 9, September 1989, pp. 112 - 115.
853. Wiener, Editor, *Focus On Analysis and Design*, SIGS Publications, Inc., New York, New York, 1991.

854. Wile and D.G. Allard, "Worlds: an Organizing Structure for Object-Bases," SIGPLAN Notices, Vol. 22, No. 1, January 1987, pp. 16 - 26.
855. Williams, "Designing the Future: The Power of Object-Oriented Programming," Apple Direct, Vol. 1, No. 4, February 1989, pp. 6 - 12.
856. Wilson, "Class Diagrams: A Tool for Design, Documentation, and Teaching," Journal of Object-Oriented Programming, Vol. 2, No. 5, January/February 1990, pp. 38 - 44.
857. Winbald, S.D. Edwards, and D.R. King, Object-Oriented Software, Addison-Wesley Publishing Company, Reading, Massachusetts, 1990.
858. Winkler, "Adding Inheritance to Ada," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 241 - 244.
859. Winston and B.K.P. Horn, Lisp, Third Edition, Addison-Wesley, Reading, Massachusetts 1989.
860. Winston, Artificial Intelligence, Second Edition, Addison-Wesley, Reading, Massachusetts, 1984.
861. Wirfs-Brock and R.E. Johnson, "Surveying Current Research in Object-Oriented Design," Communications of the ACM, Vol. 33, No. 9, September 1990, pp. 105 - 124.
862. Wirfs-Brock and B. Wilkerson, "Variables Limit Reusability," Journal of Object-Oriented Programming, Vol. 2, No. 1, May/June 1989, pp. 34 - 40.
863. Wirfs-Brock and B. Wilkerson, "Object-Oriented Design: A Responsibility-Driven Approach," OOPSLA '89 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 71 - 76.
864. Wirfs-Brock, B. Wilkerson, and L. Wiener, Designing Object-Oriented Software, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
865. Wirth, "Program Development by Stepwise Refinement," Communications of the ACM, April, 1971 pp. 221 - 227. Reprinted in E. Yourdon, Editor, Writings of the Revolution, Yourdon Press, New York, New York, 1982.
866. Wirth, Programming In Modula-2, Second Edition, Springer-Verlag, New York, New York, 1983.
867. Wisskirchen, Object-Oriented Graphics, Springer-Verlag, New York, New York, 1990.
868. Wolf, "An Object-Oriented Procedural Database for VLSI Chip Planning," Proceedings of the 23rd Design Automation Conference, CS Press, June 1986, pp. 744 - 751.
869. Wolf, "A Practical Comparison of Two Object-Oriented Languages," IEEE Software, Vol. 6, No. 5, September 1989, pp. 61 - 68.
870. Wood, "The Object of Object-Oriented Programming," NEWS/34-38, June 1986, pp. 73 - 82.
871. -C. Wu and F.-J. Wang, "An Object-Oriented Specification for Compiler," SIGPLAN Notices, Vol. 27, No. 1, January 1992, pp. 85 - 94.

872. Wu, "A Better Browser for Object-Oriented Programming," *Journal of Object-Oriented Programming*, Vol. 3, No. 4, November/December 1990, pp. 22 - 24, 26, 28.
873. Wulf, "A Case Against the GOTO," *Proceedings of the 25th ACM National Conference*, Volume 2, 1972, pp. 791-797.
874. Wybolt, "Experiences With C++ and Object-Oriented Software Development," *Software Engineering Notes*, Vol. 15, No. 2, April 1990, pp. 31 - 39.
875. Yeh, Editor, *Current Trends In Programming Methodology*, Prentice Hall, Englewood Cliffs, New Jersey, 1977.
876. Yehudai, "Data Abstraction: Types vs. Objects," *Ada Letters*, Vol. II, No. 2, September, October 1982, pp. 46 - 48.
877. Yonezawa and M. Tokoro, Editors, *Object-Oriented Concurrent Programming*, The MIT Press, Cambridge, Massachusetts, 1987.
878. Yonezawa, J.-P. Briot, and E. Shibayama, "Object-Oriented Concurrent Programming in ABCL/1," *OOPSLA '86 Conference Proceedings*, special issue of *SIGPLAN Notices*, Vol. 21, No. 11, November 1986, pp. 258 - 268.
879. Young, R.N. Taylor, and D.B. Troup, "Software Environment Architectures and User Interface Facilities," *IEEE Transactions on Software Engineering*, Vol. 14, No. 6, June 1988, pp. 697 - 708.
880. Young, "Two Models of Object-Oriented Programming and the Common Lisp Object System," *SIGPLAN Notices*, Vol. 27, No. 4, April 1992, pp. 27 - 36.
881. Yourdon and L.L. Constantine, *Structured Design: Fundamentals of a Discipline of Computer Program and Systems Design*, Prentice Hall, Englewood Cliffs, New Jersey, 1979.
882. Yourdon, Editor, *Classics in Software Engineering*, Yourdon Press, New York, New York, 1979.
883. Yourdon, "Auld Lang Syne," *Byte*, Vol. 15, No. 10, October 1990, pp. 257 - 258, 260, 262, 264.
884. Yourdon, *Object-Oriented System Design: An Integrated Approach*, Prentice Hall, Englewood Cliffs, New Jersey, 1994.
885. Zaniolo, "Object-Oriented Programming in Prolog," *1984 International Symposium on Logic Programming*, Atlantic City, New Jersey, February 1984, IEEE Computer Society Press, Washington, D.C., pp. 265 - 270.
886. Zdonik and D. Maier, "Fundamentals of Object-Oriented Databases," *Readings in Object-Oriented Database Systems*, Morgan Kaufmann Publishers, Inc. San Mateo, California, 1990, pp. 1 - 32.
887. Zdonik and D. Maier, Editors, *Readings in Object-Oriented Database Systems*, Morgan Kaufmann Publishers, Inc. San Mateo, California, 1990.

888. Zdonik, "Why Properties are Objects or Some Refinements of 'is-a'," Proceedings of the 1986 Fall Joint Computer Conference, IEEE Catalog Number 86CH2345-7, IEEE Computer Society Press, Washington, D.C., 1986, pp 41 - 47.
889. Zelkowitz, A.C. Shaw, and J.D. Gannon, Principles of Software Engineering and Design, Prentice Hall, Englewood Cliffs, New Jersey, 1979.
890. Ziegler, N. Allegre, R. Johnson, and J. Morris, "Ada for the Intel 432 Microcomputer," IEEE Computer, Vol. 14, No. 6, June 1981, pp. 47 - 56.
891. Zimmer, Abstraction for Programmers, McGraw-Hill, New York, New York, 1985.

OBJECT-ORIENTED TESTING

1. Binder, R.V.: Design for Testability in Object-Oriented Systems. Comm. of the ACM, 37(1994)9, pp. 87-101.
2. Overbeck, J.: Testing Object-Oriented Software - State of the Art and Research Directions. Proc. of the 1st EuroSTAR, London, October 25-28, 1993, pp. 245-270.

OVERVIEW

1. Cote, P.; Bourque, P.; Oligny, S.; Rivard, N.: Software Metrics: An Overview of Recent Results. The Journal of Systems and Software, 8 (1988), pp. 121-131.
2. DeGrace, P.; Stahl, L.H.: The Olduvai Imperative -- CASE and the State of the Software Engineering Practice. Yourdon Press, New Jersey, 1993.
3. Dumke, R.: Metric based software development (german). Vieweg Verlag, Braunschweig Wiesbaden, 1992.
4. Dumke, R.: Software Measures (german). Research Report, TH Magdeburg, January 1985.
5. Dumke, R.: Software measurement (german). Research Report IRB-001/91,142 S., TU Magdeburg, August 1991.
6. Dumke, R.: Software measurement -- foundations and objectivs (german). Wiss. Zeitschrift der TU Magdeburg, 36(1992)1, pp.~76-87.
7. Dumke, R.: Software metrics (german). Research Report, TU Magdeburg, July 1988.
8. Fenton, N.E.: Software Metrics -- A Rigorous Approach. Chapman & Hall, London, 1991.
9. Gilb, T.: Software Metrics. Winthrop Publishers, Inc. 1977.
10. Hamer, P.; Frewin, G.: Software metrics -- a critical overview. in: The software development process, State of the Art Report 13:2, Pergamon Infotech Limited, Maidenhead, England, 1985, pp. 61-82.
11. Harrison, W.: Bibliography on Software Complexity Metrics. SIGPLAN Notices, 19(1984)2, pp. 17-27.
12. Ince, D.: Software Metrics. in: Kitchenham; Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 27-62.

13. Ince, D.: Software metrics: introduction. Information and Software Technology, 32(1990)4, pp. 297-303.
14. Jannasch, H.: Evaluation of software quality in the phases of the software life cycle with the application of tools (german). Ph. Dissertation, Berlin 1991.
15. Measures and Metrics for Software Development. NASA, SEL-83-002, Goddard Space Flight Center, Greenbelt, Maryland, March 1984.
16. Roche, J.M.: Software Metrics and Measurement Principles. Software Engineering Notes, 19(1994)1, pp. 76-84.
17. Rubey, R.J.: Software Metrics. Tutorial on the Second International Conference on Software Quality, Research Triangle Park, NC, October 1992.
18. Shepperd, M.: Software Engineering Metrics. Vol. I, McGraw Hill Publ., 1993.
19. Thaller, G. E.: Software Metrics (german), Heise Publisher, 1994.
20. Zuse, H.: Measurement based analysis of statistical software complexity measures (german). TU Berlin, Ph. Dissertation, 1985.
21. Zuse, H.: Software Complexity - Measures and Methods. De Gruyter Publisher, Berlin, New York, 1991.
22. Zuse, H.: Software Metrics -- State of the Art. Lecture of the Workshop of the Software Metrics Group of the Society of Computer Science Germany, Technical University of Magdeburg, April 23, 1993.

PERFORMANCE MEASUREMENT

1. Card, S.K.; Moran, T.P.; Newell, A.: The Keystroke-Level Model for User Performance Time with Interactive Systems. Comm. of the ACM, 23(1980)7, pp. 396-410.
2. Smith, C.U.: Performance Engineering of Software Systems. Addison Wesley, 1990 .
3. Woodside, C.M.; Hagos, E.M.; Neron, E.; Buhr, R.J.A.: The CAEDE Performance Analysis Tool. Ada Letters, Spring, 11(1991)3, pp. 125-136.

PORTABILITY MEASURES

1. Bennett, K.; Bull, T.; Yang, H.: A Transformation System for Maintenance -- Turing Theory into Practice. Proceedings of the conference on Software Maintenance, Orlando, Nov. 9-12 1992, pp. 146-155.
2. Stallbohm, U.: A contribution to the problem solving of software portability (german). Ph. Diss., TH Aachen, 1982.
3. Tanaka, M.: A Study of Portability Problem and Evaluation. Proceedings of the Conference on Software Maintenance, Orlando, Nov. 9-12 1992, pp. 90-95.

PROCESS ANALYSIS AND MANAGEMENT

1. Balzert, H.: CASE - Systems and Tools (german). BI Wissenschaftsverlag, Mannheim Wien Zuerich, 1989.
2. Duncon, A.S.: Software Development Productivity Tools and Metrics. Proceedings of the 10th International Conference on Software Engineering, April~11-15, Singapore, 1988, pp. 41-48.
3. Heitkoetter, U.; Helling, B.; Nolte, H.; Kelly, M.: Design metrics and aids to their automation collection. Information and Software Technology, 32(1990)1, pp. 79-87.
4. Jacob, P.; Cahill, T.: A Genenric Metrication Tool. UL-CSIS-92-11, University of Lamerick, March 1992.
5. Liao, Y.; Cohen, D.: A Specificational Approach to High Level Program Monitoring and Measuring. IEEE Transactions on Software Engineering, 18(1992)11, pp. 969-978.
6. Maintenance Tools. IEEE Software, May 1990, pp.~59-66.
7. Oman, P.W.; Cook, C.R.: Design and Code Tracebility Using a PDL Metrics Tool. The Journal of Systems and Software, 12 (1990), pp.~189-198.
8. Performance Tools. IEEE Software, May 1990, S.21-30.
9. Rubin, H.A.: Macro-Estimation of Software Development Parameters: The ESTIMACS System. Proceedings of the 4th Conference of Software Engineering, 1979, pp. 109-118.
10. Saran, R.; Niech, L.E.: Establishing a Software Metrics Program at ADP . Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 276-283.
11. Schroeder, A.: Integrated Program Measurement and Documentation Tools. Proceedings of the 7th International Conference on Software Engineering, Orlando, Florida, March 1984, pp. 304-313.
12. Singh, J.; Iyer, R.K.: MEASURE: An Integrated Data-Analysis And Model-Identification Facility. University of Illonois, Urbana, January 1990.
13. The Digital COHESION Environment for CASE. Digital Press, 1990.
14. The Battlemap Analysis Tool, in: The Outlook. McCabe & Associates Newsletter, Columbia, MD, Third Quarter 1992.
15. The Digital Guide to Software Development. Digital Press, 1989.

PROGRAM ANALYSIS

1. Batson, A.P.; Brundage, R.E.: Segment Size and Lifetimes in Algol 60 Porgrams. Comm. of the ACM, 20(1977)1, pp. 36-44.
2. Berry, R.E.; Meekings, B.A.E.: A Style Analysis of C Programs. Comm. of the ACM, 28(1985)1, pp. 80-88.
3. Brown, P.J.: SCAN: A Simple Conversational Programming Language for Text Analysis. Computers and the Humanities, 6(1987)4, pp. 223-227.

4. Clark, D.W.: Measurement of Dynamic List Structure Use in Lisp. *IEEE Transactions on Software Engineering*, 5(1979)1, pp. 51-59.
5. Cook, R.P.; Lee, I.: A Contextual Analysis of Pascal Programs. *Software -- Practice and Experience*, 12(1982), pp. 195-203.
6. Eick, S.G.; Steffen, J.L.; Sommer, E.E.: Seesoft -- A Tool For Visualizing Line Oriented Software Statistics. *IEEE Transactions on Software Engineering*, 18(1992)11, pp. 957-968.
7. Elshoff, J.L.: A Numerical Profile of Commercial PL/I Programs. *Software -- Practice and Experience*, 6(1976), pp. 505-525.
8. Knuth, D.E.: An empirical study of FORTRAN programs. *Software -- Practice and Experience*, 1(1971)1, pp. 105-133.
9. Redish, K.A.; Smyth, W.F.: Program Style Analysis: A Natural By-Product of Program Compilation. *Comm. of the ACM*, 19(1986)2, pp. 126-133.
10. Ripley, G.D.; Griswold, R.E.: Tools for the Measurement of Snobol4 Programs. *SIGPLAN Notices*, 10(1975)5, pp. 36-52.
11. Robinson, S.K.; Torsun, I.S.: An empirical analysis of FORTRAN programs. *The Computer Journal*, 19(1976)1, pp. 56-62.
12. Saal, H.J.; Weiss, Z.: An Empirical Study of APL Programs. *Computer Languages*, 2(1977), pp. 47-49.
13. Salvadori, A.; Gordon, J.; Capstick, C.: Static Profile of COBOL Programs. *SIGPLAN Notices*, (1975) August, pp. 20-33.

PROGRAMMERS PRODUCTIVITY MEASURES

1. Abdel-Hamid, T.K.: Dynamics of Software Project Staffing: A System Dynamics Based Simulation Approach. *IEEE Transactions on Software Engineering*, 15(1989)2, pp. 109-119.
2. Boundy, D.: A taxonomy of programmers. *Software Engineering Notes*, 16(1991)4, pp. 23-30.
3. Esterling, B.: Software Manpower Costs: A Model. *Datamation*, March 1980, pp. 164-170.
4. Faidhi, J.A.W.; Robinson, S.K.: Programmer Experience-Level Indicators. *The Computer Journal*, 30(1987)1, pp. 52-62.
5. Gill, G.K.; Kemerer, C.F.: Cyclomatic Complexity Density and Software Maintenance Productivity. *IEEE Transactions on Software Engineering*, 17(1991)12, pp. 1284-1288.
6. Jones, T.C.: Measuring programming quality and productivity. *Tutorial -- Programming Productivity: Issues for the Eighties*, IEEE Computer Society, ISBN 0-8186-0681-9, 1986, pp. 10-34.
7. Kadary, V.: On Application of Earned Value Index to Software Productivity Metrics in Embedded Computer Systems. *Proc. on the 6th European Conference on Computer Systems*

and Software Engineering (CompEuro'92), The Hague, May 4-8 1992, Netherlands, pp. 666-670.

8. Tausworthe, R.C.: Information Models of Software Productivity: Limits on Productivity Growth. The Journal of Systems and Software, 19(1992)2, pp. 185-201.

PSEUDOCODE MEASURES

1. Reynolds, R.G.: Metric-based reasoning about pseudocode design in the partial metric system. The Journal of Systems and Software, 27(1987)9, pp. 497-502.
2. Reynolds, R.G.: Metrics to Measure the Complexity of Partial Programs. The Journal of Systems and Software, (1984)4, pp. 75-91.
3. Reynolds, R.G.: PMS: An Inference System to Monitor the Stepwise Refinement of ADA Pseudocode. IEEE Expert 1, (Winter 1986), pp. 43-49.
4. Reynolds, R.G.: The partial metrics system: modeling stepwise refinement process using partial metrics. Comm. of the ACM, 30(1987)11, pp. 956-963.

REAL-TIME PROGRAMMING MEASUREMENT

1. Cook, C.R.; Roesch, A.: Real-Time Software Metrics. The Journal of Systems and Software, 24(1994)3, pp. 223-237.

REGRESSION MODELS

1. Khoshgoftaar, T.M.; Bhattacharya, B.B.; Richardson, G.D.: Predicting Software Errors, During Development, Using Nonlinear Regression Models: A Comparative Study. IEEE Transactions on Reliability, 41(1992)3, pp. 390-395.
2. Khoshgoftaar, T.M.; Munson, J.C.: Predictive Modeling Techniques of Software Quality from Software Measures. IEEE Transactions on Software Engineering, 18(1992)11, pp. 979-987.

RELIABILITY MEASUREMENT

1. Abdel-Ghaly, A.A.; Chan, P.Y.; Littlewood, B.: Evaluation of Competing Software Reliability Predictions. IEEE Transactions on Software Engineering, 12(1986)2, pp. 950-967.
2. Baker, C.T.: Effects of Field Service on Software Reliability. IEEE Transactions on Software Engineering, 14(1988)2, pp. 254-258.
3. Becker, G.: Software Reliability (Softwarezuverlaessigkeit). Walter de Gruyter, Berlin New York, 1989.
4. Bloomfield, R.E.; Froome, P.K.D.: Aspects of the Licensing and Assessment of Highly Dependable Computer Systems. in: Kitchenham; Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 243-262.
5. Brocklehurst, S.; Chan, P.Y.; Littlewood, B.; Snell, J.: Adaptive Software Reliability Modeling. 263-278.

6. Cheung, R.C.: A User-Oriented Software Reliability Model. IEEE Transactions on Software Engineering, 6(1980)2, pp. 118-125.
7. Chillarege, R.; Bhandari, S.; Chaar, J.K.; Halliday, M.J.; Moebus, D.S.; Ray, B.K.; Wong, M.: Orthogonal Defect Classification -- Concepts for In-Process Measurements. IEEE Transactions on Software Engineering, 18(1992)11, pp. 943-956.
8. Currit, P.A.; Dyer, M.; Mills, H.D.: Certifying the Reliability of Software. IEEE Transactions on Software Engineering, 12(1986)1, pp.~3-11.
9. Dyer, M.: The Cleanroom Software Development Process. in: Kitchenham; Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 1-26.
10. Ehrlich, W.K.; Lee, S.K.; Molisani, R.H.: Applying Reliability Measurement: A Case Study. IEEE Software, March 1990, pp. 56-64.
11. Ehrlich, W.K.; Stampfel, J.P.; Wu, J.R.: Application of Software Reliability Modeling to Product Quality and Test Process. Proceedings of the 12th International Conference on Software Engineering, March~26-30, Nice, France, 1990, pp. 108-116.
12. Everett, W.W.: An "Extended Execution Time" Software Reliability Model. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, 1992, pp. 4-13.
13. Everett, W.W.; Musa, J.D.: A software reliability engineering practice. IEEE Computer, March 1993, pp. 77-79.
14. Fenton, N.; Littlewood, B.: Software Reliability and Metrics. Elsevier Applied Science, London New York, 1991.
15. Hevner, A.R.; Vagoun, T.; Lemmon, D.: Quality Measurement in the Cleanroom Development Process. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 80-87.
16. Lew, K.S.; Dillon, T.S.; Forward, K.E.: Software Complexity and its Impact on Software Reliability. IEEE Transaction on Software Engineering, 14(1988)11, pp. 1645-1655.
17. Mellor, P.: Modelling the Support Process. in: Kitchenham; Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp.~279-288.
18. Moeller, K.: Error estimation for software products in the evolutionary development of versions (german). Informationstechnik, Munich, 31(1989)4, pp. 220-265.
19. Munson, J.C.: Software Faults, Software Failures and Software Reliability Modeling. Proc. of the 11th Conference on Software Evolution, Models and Metrics, September 7-9, Dublin, Ireland, Section 2.
20. Musa, J.D.: A Theory of Software Reliability and Its Application. IEEE Transactions on Software Engineering, 1(1975)3, pp. 312-327.
21. Musa, J.D.; Ackerman, A.F.: Quantifying Software Validation: When to Stop Testing? IEEE Software, May 1989, pp.~19-27.

22. Musa, J.D.; Iannino, A.; Okumoto, K.: Software Reliability -- Measurement, Prediction, Application. McGraw-Hill International Editions, 1987.
23. Petrova, E.; Veevers, A.: Role of non-stochastic-based metrics in quantification of software reliability. Information and Software Technology, 32(1990)1, pp. 71-78.
24. Reibman, A.L.; Veeraraghavan, M.: Reliability Modeling: An Overview for System Designers. IEEE Computer, April 1991, pp. 49-57.
25. Rubey, R.J.; Brewer, A.C.: Use of a Simple Reliability Model as a Software Test Metric. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 156-160.
26. Shafto, S.: Predictive Modeling of Software Reliability: A Multidimensional Approach. Proceedings of the Annual Oregon WEorkshop on Software Metrics, April 10-12, 1993, Silver Falls, Oregon.
27. Siefert, D.M. : Software Reliability Handbook. Proceedings of the International Software Quality Conference, Dayton, Ohio, 1991, pp. 184-189.

REQUIREMENT ANALYSIS

1. Davis, A. et al: Identifying and Measuring Quality in a Software Requirements Specification. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 141-152.
2. Ebert, C.; Oswald, H.: Complexity Measures for the Analysis of Specifications of (Reliability Related) Computer Systems. Proceedings of the IFAC SAFECOMP, London, 1991.
3. Ebert, C.; Oswald, H.: Improving Specification with Complexity Measures. Proceedings of the 3rd Int. Workshop on Rapid System Prototyping, June 23-25, Research Triangle Park, NC, 1992.
4. Farbey, B.: Software quality metrics: considerations about requirements and requirement specifications. Information and Software Technology, 32(1990)1, pp. 60-64 and the IEEE 982 Standard.
5. Knoell, H.-D.; Suk, W.: A model of user oriented software quality assurance for the requirement analysis of commercial software systems (german). Informatik - Forschung und Entwicklung, (1991)6, pp. 28-35.
6. Ramamoorthy, C.V.; Tsai, W., Yamaura, T.; Bhide A.: Metrics Guided Methodology. COMPSAC 85, pp. 111-120.
7. Werthmann, T.: Evaluations of user interfaces in relating to economical aspects (german). Master's Thesis, TU Magdeburg, 1993.

REUSABILITY MEASURES

1. Beck, J.; Eichmann, D.: Program and Interface Slicing for Reverse Engineering. Proceedings of the 15th International Conference on Software Engineering, May 17-21, Baltimore, 1993, pp. 509-517.

2. Bieman, J.M.: Deriving Measures of Software Reuse in Object Oriented Systems. in: Denvir et. al.: Formal Aspects of Measurement. Springer Verlag, 1992, pp. 63-83.
3. Caldiera, G.; Basili, V.R.: Identifying and Qualifying Reusable Software Components. IEEE Computer, February 1991, pp. 61-70.
4. Cimitile, A.; Carlini, U.: Rverse Engineering: Algorithms for Program Graph Production. Software - Practice and Experience, 21(1991)5, pp. 519-537.
5. Kneffel, I.: Analysis of the enterprise software base in the point of view to the economical based replacement (german). Master's Thesis, TU Magdeburg, 1992.
6. Mueller, H.A.; Moehr, J.R.; McDaniel, J.G.: Applying Software Re-engineering Techniques to Health Information Systems. University of Victoria, Canada, DCS-138-IR, July 1990.
7. Poulin, J.S.; Caruso, J.M.: Determining the Value of a Corporate Reuse Proram. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 16-27.
8. Prieto-Diaz, R.: Implementing Faceted Classification for Software Reuse. Comm. of the ACM, 34(1991)5, pp. 88-97.
9. Prieto-Diaz, R.; Freeman, P.: Classifying Software for Reusability. IEEE Software, January 1987.

Reuse Bibliography

1. Adelson and E. Soloway, "The Role of Domain Experience in Software Design," IEEE Transactions on Software Engineering, Vol. SE-11, No. 11, November 1985, pp. 1351 - 1360.
2. Alexandridis, "Adaptable Software and Hardware: Problems and Solutions, " IEEE Computer, Vol. 19, No. 2, February 1986, pp. 29 - 39.
3. Al-Haddad, K.M. George, and M.H. Samadzadeh, "Approaches to Reusability in C++ and Eiffel," Journal of Object-Oriented Programming, Vol. 4, No. 5, September 1991, pp. 34, 36 - 41, 43 - 45.
4. Allen and P.L. Holtzman, "Simplifying the Construction of Domain-Specific Automatic Programming Systems: The NASA Automated Software Development Workstation Project," Proceedings of the Space Operations Automation and Robotics Workshop, Houston, Texas, 1987, pp. 407 - 410.
5. Arango, "Evaluation of a Reuse-Based Software Construction Technology," Proceedings of the Second IEE/BCS Conference: Software Engineering 88, London, United Kingdom, 1988, pp. 85 - 92.
6. Arango, "Domain Analysis: From Art to Engineering Discipline," Proceedings of the Fifth International Workshop On Software Specification and Design (Software Engineering Notes, Vol. 14, No. 3), May 1989, pp. 152 - 159.

7. Arico and A. Gargaro, "Disciplined Reusable Ada Programming for Real-Time Applications," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 443 - 455.
8. Atkinson, Object-Oriented Reuse, Concurrency and Distribution, Addison-Wesley Publishing Company, Reading, Massachusetts, 1991.
9. Bailey and V.R. Basili, "Software Reclamation: Improving Post-Development Reusability," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 477 - 499.
10. Bardin and C. Thompson, "Using the Re-Export Paradigm to Build Composable Ada Software Components," Ada Letters, Vol. 8, No. 2, March/April 1988, pp. 39 - 54.
11. Barnes and T. Bollinger, "Making Reuse Cost Effective," IEEE Software, Vol. 8, No. 1, January 1991, pp. 13 - 24.
12. Barstow, "Domain-Specific Automatic Programming," IEEE Transactions on Software Engineering, Vol. SE-11, No. 11, November 1985, pp. 1321 - 1336.
13. Basili, R.W. Selby, Jr. and T.-Y. Phillips, "Metric Analysis and Data Validation Across FORTRAN Projects," IEEE Transactions on Software Engineering, Vol. SE-9, No. 6, November 1983, pp. 652-663.
14. R. Basili, Editor, Tutorial on Models and Metrics for Software Management and Engineering, IEEE Computer Society Press (catalog number EHO-167-7), New York, New York, 1980.
15. Bassett, "Frame-Based Software Engineering," IEEE Software, Vol. 4, No. 4, July 1987, pp. 9 - 16.
16. Batory, J.R. Barnett, J. Roy, B.C. Twichell, and J.F. Garza, Construction of File Management Systems From Software Components, Technical Report TR-88-36 REV, University of Texas, Austin, Texas, October 1988.
17. Batory, V. Singhal, J. Thomas, S. Sasari, B. Geraci, and M. Sirkin, "The Gen Voca Model of Software-System Generators," IEEE Software, Vol. 11, No. 5, September 1994, pp. 89 - 94.
18. Beane, N. Giddings, and J. Silverman, "Quantifying Software Designs," Proceedings of the Seventh International Conference on Software Engineering, 1984, pp. 314 - 322.
19. Behrens, "Measuring the Productivity of Computer Systems Development," IEEE Transactions on Software Engineering, Vol. SE-9, No. 6, November 1983, pp. 648 - 651.
20. Berard, An Object-Oriented Design Handbook for Ada Software, EVB Software Engineering, Inc., Frederick, Maryland, 1985.
21. Berard, "Creating Reusable Ada Software," Proceedings of the National Conference on Software Reusability and Maintainability, September, 1986.
22. Berard, Creating Reusable Ada Software, (Course Notes), EVB Software Engineering, Inc., Frederick, Maryland, 1987.

23. Berard, "Software Reusability Cannot Be Considered in a Vacuum," Digest of Papers COMPCON, Spring 1987, IEEE Catalog Number 87CH2409-1, Computer Society Order Number 764, Computer Society Press of the IEEE, Washington, D.C., pp. 390 - 393.
24. Berlin, "When Objects Collide: Experiences With Reusing Multiple Class Hierarchies," OOPSLA/ECOOP '90 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 25, No. 10, October 1990, pp. 181 - 193.
25. Biggerstaff and A.J. Perlis, "Foreword: Special Issue on Software Reusability ," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1987, pp. 474 - 477.
26. Biggerstaff and A.J. Perlis, Software Reusability, Volume 1: Concepts and Models, Addison-Wesley, Reading, Massachusetts, 1989.
27. Biggerstaff and A.J. Perlis, Software Reusability, Volume 2: Applications and Experience, Addison-Wesley, Reading, Massachusetts, 1989.
28. Biggerstaff and C. Richter, "Reusability Framework, Assessment, and Directions," IEEE Software, Vol. 4, No. 2, March 1987, pp. 41 - 49.
29. Biggerstaff, "Design Recovery for Maintenance and Reuse," IEEE Computer, Vol. 22, No. 7, July 1989, pp. 36 - 49.
30. Boehm, J.R. Brown and M. Lipow, "Quantitative Evaluation of Software Quality," Tutorial on Models and Metrics for Software Management and Engineering, ed. V.R. Basili, IEEE Computer Society Press, 1980, New York, New York, pp. 218 - 231.
31. Boehm, Software Engineering Economics, Prentice-Hall, Englewood Cliffs, New Jersey, 1981.
32. Reusable Software Catalog (CDRL 109), Boeing Aerospace Company, Seattle, Washington, 1986.
33. Bollinger and B.H. Barnes, "Reuse Rules: An Adaptive Approach to Reusing Ada Software," Proceedings of AIDA-88, George Mason University, Fairfax, Virginia, November 1988, pp. 14-1 - 14-8.
34. Bollinger and S.L. Pfleeger, " The Economics of Reuse: Issues and Alternatives," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 436 - 447.
35. Booch, Software Components With Ada, Benjamin/Cummings, Menlo Park, California, 1987.
36. Bowen, "Are Current Approaches Sufficient for Measuring Software Quality?," Tutorial on Software Quality Assurance Practical Approach, ed. T.S. Chow, IEEE Computer Society Press, New York, New York, 1985, pp. 62-69.
37. Brogida, J. Mylopoulos, and H.K.T. Wong, "Generalization/Specialization As a Basis for Software Specifications," in On Conceptual Modeling, Springer-Verlag, New York, New York, 1984, pp. 87 - 117.
38. Brown and R.B. Quanrud, "The Generic Architecture Approach to Reusable Software," Proceedings of the Sixth National Conference on Ada Technology, March 14-18, 1988,

U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 390 - 394.

39. Bruns and C. Potts, Domain Modeling Approaches to Software Development, Technical Report STP-186-88, Microelectronics and Computer Technology Corporation, Austin, Texas, June 1988.
40. Burnham, R.D. Geradi, P. Ho, and H.F. Joiner, "Analysis of Software Reuse on AFATDS Concept Evaluation," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 87 - 91.
41. Burton, R.W. Aragon, S.A. Bailey, K.D. Koehler, and L. A. Mayes, "The Reusable Software Library," IEEE Software, Vol. 4, No. 4, July 1987, pp. 25 - 33.
42. Caldiera and V.R. Basili, "Identifying and Qualifying Reusable Software Components," IEEE Computer, Vol. 24, No. 2, February 1991, pp. 61 - 70.
43. Calvano and J.A. McCall, "A Framework for the Measurement of Software Technology," Tutorial on Software Quality Assurance A Practical Approach, ed. T.S. Chow, IEEE Computer Society Press, New York, New York, 1985, pp. 46-52.
44. Carstensen, "A Real Example of Reusing Ada Software," Proceedings of the Second National Conference on Software Reusability and Maintainability, National Institute for Software Quality and Productivity, Washington, D.C., March 1987, pp. B-1 to B-19.
45. Chan, "Lessons in Software Reusability in Large Complex Software Systems," Proceedings of the Conference on Software Reusability and Portability, National Institute for Software Quality and Productivity, Washington, D.C., September 16-17 1987, pp. B-1 to B-7.
46. Cheatham, "Reusability Through Program Transformations," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 589 - 594.
47. Chen and W. Sobkiw, "Reusable Subsystems From a High Performance Ada Communication System," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 411 - 418.
48. Chen and W. Sobkiw, "Binding As a Mechanism to Support Reusability in a Distributed Ada Communications System," Proceedings of the Sixth Washington Ada Symposium, June 26-29, 1989, pp. 155 - 162.
49. Clapp, "Software Reusability: A Management View," Proceedings, Compsac84 Conference, Silver Spring, Maryland: IEEE Computer Society Press, Washington, DC, 1984, pp. 479 - 480.
50. Coggins, "Design Criteria for C++ Libraries," Proceedings of the C++ Conference, San Francisco, California, April 1990, USENIX Association, Berkeley, California, 1990, pp. 25 - 36.
51. Conn, "The Ada Software Repository," Digest of Papers COMPCON, Spring 1987, IEEE Catalog Number 87CH2409-1, Computer Society Order Number 764, Computer Society Press of the IEEE, Washington, D.C., pp. 372 - 375.
52. Cox, Object Oriented Programming: An Evolutionary Approach, Addison-Wesley, Reading, Massachusetts, 1986.

53. Curtis, S.B. Sheppard and P. Milliman, "Third Time Charm: Stronger Prediction of Performance by Software Complexity Measures," Tutorial on Programming Productivity: Issues for the Eighties, Edited by C. Jones, IEEE Computer Society Press, New York, New York, 1981, pp. 81 - 85.
54. D'Ippolito, "Using Models In Software Engineering," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 256 - 265.
55. Denning, "Throwaway Programs," Communications of the ACM, Vol. 24, No. 2, February 1981, pp. 259 - 260.
56. Devanbu and R. Bachman, OOPSLA Workshop on Domain Modeling in Software Engineering, New Orleans, Louisiana, 1989.
57. Devanbu, P. Selfridge, B. Ballard, and R. Brachman, "Steps toward a Knowledge Based Software Information System," Proceedings International Joint Conference on Artificial Intelligence, Detroit, Michigan, 1989, pp. 110 - 115.
58. Devanbu, P. Selfridge, and R. Brachman, "Inference in Support of Retrieval for Re-Use in Large Software Systems," IEEE/SPS Workshop on Software Re-Use, Indialantic, Florida, 1989.
59. Dlugosz, "Libraries With Class," Byte, Vol. 16, No. 2, February 1991, pp. 164 - 166, 168.
60. -E. Doberkat, "Tangraml -- A Program Description Language for Ada," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 390 - 403.
61. Department of Defense, Department of Defense Requirements for High Order Computer Programming Languages: "Steelman", NTIS Order Number ADA059444, 1978.
62. Dusink and P. Hall, Editors, Software Re-Use, Utrecht 1989, Springer-Verlag, New York, New York, 1991.
63. Dusink and J. van Katwijk, "Reflections on Reusable Software and Software Components," Ada Components: Libraries and Tools - Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Edited by S. Tafvelin, Cambridge University Press, Cambridge, U.K., pp. 113 - 126.
64. O. Ejiogu, "The Critical Issues of Software Metrics, Part 0: Perspectives on Software Metrics," SIGPLAN Notices, Vol. 22, No. 3, March 1987, pp. 59 - 64.
65. Embley and S.N. Woodfield, "A Knowledge Structure for Reusing Abstract Data Types in Ada Software Production," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 27 - 34.
66. Fafchamps, "Organizational Factors and Reuse," IEEE Software, Vol. 11, No. 5, September 1994, pp. 31 - 41.

67. Ferber, "Computational Reflection In Class Based Object-Oriented Languages," OOPSLA '89 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 24, No. 10, October 1989, pp. 317 - 326.
68. Fischer, "Cognitive View of Reuse and Redesign," IEEE Software, Vol. 4, No. 4, July 1987, pp. 60 - 72.
69. Fischer, A. Girgensohm, K. Nakakoji, and D. Redmiles, "Supporting Software Designers with Integrated Domain-Oriented Design Environments," IEEE Transactions on Software Engineering, Vol. 18, No. 6, June 1992, pp. 511 - 522.
70. Frakes and P.B. Gandel, "Representation Methods for Software Reuse," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 302 - 314.
71. Frakes and S. Isoda, "Success Factors of Systematic Reuse," IEEE Software, Vol. 11, No. 5, September 1994, pp. 15 - 18.
72. W.B Frakes and B.A. Nejme, "An Information System for Software Re-Use," Proceedings of the Tenth Minnowbrook Workshop on Software Re-Use, 1987, pp. 142 - 151.
73. Frakes and B.A. Nejme, "Software Reuse Through Information Retrieval," Proceedings of the Twentieth Hawaii International Conference on System Sciences, Kailua-Kona, Hawaii, 1987, pp. 530 - 535.
74. Frakes and T.P. Pole, "An Emperical Study of Representation Methods for Reusable Software Components," IEEE Transactions on Software Engineering, Vol. 20, No. 8, August 1994, pp. 617 - 630.
75. N. Frankowski. "Why Programs Built from Reusable Software Should be Single Paradigm," Proceedings of the STARS Reusability Workshop, March 24-27, 1986.
76. Freeman, "Reusable Software Engineering: Concepts and Research Directions," Tutorial: Software Reusability, IEEE Computer Society Press, Washington, DC, 1987, pp. 10 - 23.
77. "ObjectDrawing -- Reusing the FreeDrawing Class," Journal of Object-Oriented Programming, Vol. 3, No. 1, May/June 1990, pp. 9 - 16.
78. Fujino, "Software Factory Engineering: Today and Future," Proceedings of the 1987 Fall Joint Computer Conference, Dallas, Texas, 1987, pp. 262 - 270.
79. Gamma, R. Helm, R. Johnson, and J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Publishing Company, Reading, Massachusetts, 1995.
80. Gargaro and T.L. Pappas, "Reusability Issues and Ada," IEEE Software, Vol. 4, No. 4, July 1987, pp. 43 - 51.
81. Gautier, "A Language for Describing Ada Software Components," Ada Components: Libraries and Tools - Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Edited by S. Tafvelin, Cambridge University Press, Cambridge, U.K., pp. 75 - 84.

82. Gilb, Software Metrics, Winthrop Publishers, Inc., Cambridge, Massachusetts, 1977.
83. Gilroy, E.R. Comer, J.K. Grau, and P.J. Merlet, Impact of Domain Analysis on Reuse Methods, Final Report C04-087LD-0001-00, U.S. Army Communications-Electronics Command, Ft. Monmouth, New Jersey, November 1989.
84. Gorlen, S. M. Orlow, and P.S. Plexico, Data Abstraction and Object-Oriented Programming in C++, John Wiley and Sons, New York, New York, 1990.
85. Gorlen, "An Object-Oriented Class Library for C++ Programmers," Software: Practice and Experience, Vol. 17, No. 12, December 1987, pp. 899 - 922.
86. Grady and D.L. Caswell, Software Metrics: Establishing a Company-Wide Program, Prentice-Hall, Englewood Cliffs, New Jersey, 1987.
87. Guindi, W.M. McCracken, and S. Rugaber, "Reuse and the Software Life-Cycle," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 463 - 468.
88. Hailpern and H. Ossher, "Extending Objects to Support Multiple Interfaces and Access Control," IEEE Transactions on Software Engineering, Vol. 16, No. 11, November 1990, pp. 1247 - 1257.
89. Halstead, Elements of Software Science, Elsevier, North-Holland, New York, 1977.
90. Harris and W.L. Johnson, "Sharing and Reuse of Requirements Knowledge," Proceedings of the Sixth Annual Knowledge-Based Software Engineering Conference (KBSE-91), September 1991, pp. 65 - 77.
91. Harrison, "AdaL, An Automated Code Reuse System," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 404 - 410.
92. Henderson-Sellers and C. Freeman, "Cataloguing and Classification for Object Libraries," Software Engineering Notes, Vol. 17, No. 1, January 1992, pp. 62 - 64.
93. Henninger, "Using Iterative Refinement for Find Reusable Software," IEEE Software, Vol. 11, No. 5, September 1994, pp. 48 - 59.
94. Hess, W.E. Novak, P.C. Carroll, S.G. Cohen, R.R. Holibaugh, K.C. Kang, and A.S. Peterson, A Domain Analysis Bibliography, Special Report CMU/SEI-90-SR-3, Software Engineering Institute, Pittsburgh, Pennsylvania, 1990.
95. Hibbard, A. Hisgen, J. Rosenberg, M. Shaw, M. Sherman, Studies in Ada Style, Springer-Verlag, New York, New York, 1983.
96. Holibough, S. Cohen, K. Kang, and S. Peterson, "Reuse: Where to Begin and Why," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 266 - 277.
97. Hooper and R.O. Chester, "Software Reuse: Managerial and Technical Guidelines," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 424 - 435.

98. Hooper and R.O. Chester, "Software Reuse Progress," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 39 - 44.
99. Horowitz and J.B. Munson, "An Expansive View of Reusable Software," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 477 - 487.
100. IEEE, IEEE Standard Glossary of Software Engineering Terminology, The Institute of Electrical and Electronic Engineers, Inc., New York, 1983.
101. Institute for Electrical and Electronics Engineers, IEEE Standard Glossary of Software Engineering Terminology, IEEE Standard 610.12-1990, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1991.
102. Institute for Electrical and Electronics Engineers, IEEE Standards Collection: Software Engineering, 1993 Edition, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1993.
103. Reference Manual of the International Mathematical & Statistics Libraries , IMSL Inc., Houston, Texas, 1979.
104. Iscoe, "Domain-Specific Reuse: An Object-Oriented and Knowledge-Based Approach," in IEEE Tutorial: Software Reuse: Emerging Technology, Edited by W. Tracz, IEEE Catalog No. EH0278-2, IEEE Computer Society Press, Washington, D.C., 1988, pp. 299 - 308.
105. Jaworski, F. Hills, T.A. Durek, S. Faulk, and J.E. Gaffney, A Domain Analysis Process, Interim Report 90001-N (Version 01.00.03), Software Productivity Consortium, Herndon, Virginia, January 1990.
106. Johnson and B. Foote, "Designing Reusable Classes," Journal of Object-Oriented Programming, Vol. 1, No. 2, July/August 1988, pp. 22-35.
107. Joiner, "Economic Analysis of Software Reuse: A Model Study," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 448 - 451.
108. Jones, R.E. Bozman, and W. McIver, "The Morehouse Object-Oriented Reuse Library System," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 456 - 462.
109. Jones, Program Quality and Programmer Productivity, IBM Technical Report TR 02.764, International Business Machines Corporation, San Jose, California, January 1977.
110. Jones, "The Limits of Programmer Productivity," Proceedings of the Joint SHARE/GUIDE/IBM Application Development Symposium, October 1979, pp. 77 - 82.
111. Jones, Editor, Tutorial: Programmer Productivity: Issues for The Eighties, IEEE Catalog Number EHO186-7, Computer Society Order Number 391.
112. Jones, "Reusability in Programming: A Survey of the State of the Art," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 488-494.
113. Jones, Editor, Tutorial Programming Productivity: Issues for The Eighties, Second Edition, IEEE Computer Society Press (catalog number EHO239-4), Washington, DC, 1986.

114. Joos, "Software Reuse at Motorola," IEEE Software, Vol. 11, No. 5, September 1994, pp. 42 - 47.
115. Kaiser and D. Garlan, "Melding Software Systems from Reusable Building Blocks," IEEE Software, Vol. 4, No. 4, July 1987, pp. 17 - 24.
116. Kang, S.G. Cohen, J.A. Hess, W.E. Novak, and A.S. Peterson, Feature-Oriented Domain Analysis (FODA) Feasibility Study, Technical Report CMU/SEI-90-TR-21 (ESD-90-TR-222), Software Engineering Institute, Carnegie-Mellon University, Pittsburgh, Pennsylvania, November 1990.
117. C. Kendall and E. C. Lamb, "Management Perspectives On Programs, Programming, and Productivity," Guide 45, Atlanta, Georgia, November 1977.
118. "An Architecture of Reusability in Programming," Proceedings of the ITT Workshop on Reusability in Programming, T. Biggerstaff, Editor, ITT Programming Technology Center, Stratford, Connecticut, September 1983, pp. 1 - 3.
119. Kernighan and P.J. Plauger, The Elements of Programming Style, 2nd ed., McGraw-Hill Book Company, New York, New York, 1978.
120. Kernighan, "Reusability in the Smalltalk-80 Programming System," Proceedings of the Workshop on Reusability in Programming, ITT, Shelton, Connecticut, September 1983, pp. 235 - 239.
121. W. Kernighan, "The UNIX System and Software Reusability," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1985, pp. 488 - 518.
122. Kitaoka, "Establishing Ada Repositories for Reuse," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 315 - 323.
123. Kitaoka, "Repository Support for a Reuse Process," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 471 - 476.
124. Knuth, The Art of Computer Programming, Volume 1/Fundamental Algorithms, 2nd Ed., Addison-Wesley, Reading, Massachusetts, 1973.
125. Knuth, "Structured Programming with GOTO's", Current Trends in Programming Methodology Vol. 1, Prentice- Hall, Englewood Cliffs, New Jersey, 1977.
126. LaLonde, "Designing Families of Data Types Using Exemplars," ACM Transactions on Programming Languages and Systems, Vol. 11, No. 2, April 1989, pp. 17 - 34.
127. Lanergan and C.A. Grasso, "Software Engineering With Reusable Designs and Code," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 498 - 501.
128. Lanergan and B.A. Poynton, "Reusable Code: The Application Development Technique of the Future," Proceedings of the IBM SHARE/GUIDE Software Symposium, IBM, Monterey, California, October 1979, pp. 127 - 136.
129. Latour and C. Meadow, "Engineering Reuse Using Procedure Types," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 106 - 117.

130. Leach, "Software Metrics Analysis of the Ada Repository," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 270 - 277.
131. Leavitt, "Some Practical Experience in the Organization of a Library of Reusable Ada Units," Proceedings of the 3rd Annual National Conference on Ada Technology, March 20-21, 1985, pp. 68 - 71.
132. Ledbetter and B. Cox, "Software ICs," Byte, Vol. 10, No. 6, June 1985, pp. 307 - 315.
133. Ledgerd, Programming Proverbs, Hayden Book Company, Rochelle Park, New Jersey, 1975.
134. Lenz, H.A. Schmid, and P.F. Wolf, "Software Reuse Through Building Blocks," IEEE Software, Vol. 4, No. 4, July 1987, pp. 34 - 42.
135. Levy and K. Ripken, "Experience in Constructing Ada Programs from Non-Trivial Reusable Modules," Ada Components: Libraries and Tools - Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Edited by S. Tafvelin, Cambridge University Press, Cambridge, U.K., pp. 100 - 112.
136. Levy, "A Methodology for the Evaluation of Reusable Ada Software Libraries," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 92 - 99.
137. Lewis, S.M. Henry, D.G. Kafura, and R.S. Schulman, "An Empirical Study of the Object-Oriented Paradigm and Software Reuse," OOPSLA '91 Conference Proceedings, Special Issue of SIGPLAN Notices, Vol. 26, No. 11, November 1991, pp. 184 - 196.
138. Liberherr and I.M. Holland, "Assuring Good Style for Object-Oriented Programs," IEEE Software, Vol. 6, No. 5, September 1989, pp. 38 - 48.
139. Lim, "Effects of Reuse on Quality, Productivity, and Economics," IEEE Software, Vol. 11, No. 5, September 1994, pp. 23 - 30.
140. Lin and G.W. Ernst, "Reusability and Verification of Generics in Ada," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 98 - 105.
141. Lin and S. Harous, "Reusability and Extensibility in Ada," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 269 - 274.
142. Litvintchouk and A.S. Matsumoto, "Design of Ada Systems Yielding Reusable Components: An Approach Using Structured Algebraic Specification," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 544 - 551.
143. Lorenz, "Real World Reuse," Journal of Object-Oriented Programming, Vol. 4, No. 7, November/December 1991, pp. 35 - 39.
144. Lubars, "Code Reusability In the Large Versus Code Reusability in the Small," Software Engineering Notes, Vol. 11, No. 1, January 1986, pp. 21 - 27.
145. Lubars, Domain Analysis and Domain Engineering in IDeA, Technical Report STP-295-88, Microelectronics and Computer Technology Corporation, Austin, Texas, September 1988.

146. Luckey and F.G. DuPont, "Rapid Prototyping in Ada in the Rational Environment Emphasizing Software Reuse," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 71 - 75.
147. Mac An Airchinnigh, A. Burns, and C. Chedgey, "Reusable Units - Construction Methods and Measure," Ada Components: Libraries and Tools - Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Edited by S. Tafvelin, Cambridge University Press, Cambridge, U.K., pp. 127 - 140.
148. Mac An Airchinnigh, "Reusable Generic Packages -- Design Guidelines Based On Structural Isomorphism," Proceedings of the 3rd Annual National Conference on Ada Technology, March 20-21, 1985, pp. 132 - 144.
149. Mackey, M. Downs, J. Duffy, and J. Leege, "Achieving Reusability Through an Interactive Ada Interface Builder," Proceedings of the Seventh Washington Ada Symposium, June 25-28, 1990, pp. 313 - 329.
150. Mackey, "Reusable Ada Modules for Optimal Resource Allocation," Proceedings of the Seventh Annual National Conference on Ada Technology, March 13-16, 1989, pp. 9 - 12.
151. Margono and E.V. Berard, "A Modified Booch's Taxonomy for Ada Generic Data-Structure Components and Their Implementation," Ada Components: Libraries and Tools - Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Edited by S. Tafvelin, Cambridge University Press, Cambridge, U.K., pp. 61 - 74.
152. Masters and M.J. Kuchinski, "Software Design Prototyping Using Ada," Ada Letters, Vol. II, No. 4, January-February 1983, pp. 68 - 75.
153. Matsumoto, "Some Experiences in Promoting Reusable Software: Presentation in Higher Abstract Levels," IEEE Transaction on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 502 - 513.
154. McCabe, "A Complexity Measure," IEEE Transactions on Software Engineering, Vol. SE-2, No. 4, October 1976, pp. 243-245.
155. McCall, D. Markham, M. Stosick and R. McGindly, "The Automated Measurement of Software Quality," Tutorial on Software Quality Assurance A Practical Approach, ed. T.S. Chow, IEEE Computer Society Press, New York, New York, 1985, pp. 388-394.
156. McGregor and D.A. Sykes, Object-Oriented Software Development: Engineering Software for Reuse, Van Nostrand Reinhold, New York, New York, 1992.
157. McIlroy, "Mass Produced Software Components," Software Engineering: Report on a Conference Sponsored by the NATO Science Committee, pp. 138 - 150.
158. McIlroy, "Mass Produced Software Components," Software Engineering Concepts and Techniques, 1968 NATO Conference on Software Engineering, J.M. Buxton, P. Naur and B. Randell, Eds. 1976, pp. 88 - 98.
159. McNicholl et al, Common Ada Missile Packages (CAMP) - Volume 1: Overview and Commonality Study Results, Technical Report AFATL-TR-85-93, McDonnell Douglas Astronautics Company, St. Louis, Missouri, May 1986.

160. G. McNicholl, S. G. Cohen, and C. Palmer et al, Common Ada Missile Packages -- Phase 2 (CAMP-2) -- Volume 1: CAMP Parts and Parts Composition System, Final Report AFAL-TR-88-62, Volume 1, McDonnell Douglas Astronautics Company, St. Louis, Missouri, November 1988.
161. Mendal, "Designing for Ada Reuse: A Case Study", in the Proceedings of the IEEE Computer Society Second International Conference on Ada Applications and Environments, April 8-10, 1986, Miami Beach, FL, pages 33 - 42.
162. Meyer, "Eiffel: Programming for Reusability and Extendability," SIGPLAN Notices, Vol. 22, No. 2, February 1987, pp. 85 - 94.
163. Meyer, "Reusability: The Case for Object-Oriented Design," IEEE Software, Vol. 4, No. 2, March 1987, pp. 50 - 64.
164. Micallef, "Encapsulation, Reusability, and Extensibility in OOP Languages," Journal of Object-Oriented Programming, Vol. 1, No. 1, April/May 1988, pp. 12 - 35.
165. A. Miller, "The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information," The Psychological Review, Vol. 63, No. 2, March 1956, pp. 81 - 97. Reprinted in R.T. Mittermeir and M. Opus, "Software Bases for Flexible Composition of Application Systems," IEEE Transactions on Software Engineering, Vol. SE-13, No. 4, April 1987, pp. 440 - 460.
166. Morgan, "Configuration Management and Version Control in the Rational Programming Environment," in Ada In Industry: Proceedings of the Ada-Europe International Conference Munich 7-9 June, 1988, Cambridge University Press, Cambridge, United Kingdom, 1988, pp. 17 - 28.
167. Muralidharan and B.W. Weide, "Should Data Abstraction Be Violated to Enhance Software Reuse?," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 515 - 524.
168. Myers, Software Reliability, Principles and Practices, John Wiley and Sons, New York, New York, 1976.
169. NAG FORTRAN Library Manual, Numerical Analysis Group Inc., Oxford, UK, 1980.
170. Naur and B. Randell, Editors, Software Engineering: Report on a Conference Sponsored by the NATO Science Committee, Garmisch, Germany, October 7-11, 1968.
171. Neighbors, "Software Construction Using Components," Technical Report 160, Department of Information and Computer Sciences, University of California, Irvine, 1980.
172. Neighbors, "The Draco Approach to Constructing Software from Reusable Components," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 564 - 574.
173. Newsted, W. K. Long, J. Yeung, "The Impact of Programming Styles on Debugging Efficiency", ACM SIGSOFT Software Engineering Notes, Vol. 6, No. 5, 1981.
174. Nissen and P. Wallis, Portability and Style In Ada, Cambridge University Press, Cambridge, United Kingdom, 1984.

175. O'Connor, C. Mansour, J. Turner-Harris, and G.H. Campbell, "Reuse in Command-and-Control Systems," IEEE Software, Vol. 11, No. 5, September 1994, pp. 70 - 79.
176. Paige, "A Metric for Software Test Planning," Tutorial on Software Quality Assurance Practical Approach, Edited by T.S. Chow, IEEE Computer Society Press, New York, New York, 1985, pp. 70-75.
177. Pappas, "Literate Programming for Reusability: A Queue Package Example," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 500 - 514.
178. Parnas, P.C. Clements, and D. M. Weiss, "Enhancing Reusability with Information Hiding," ITT Proceedings of the Workshop on Reusability in Programming, 1983, pp. 240 - 247.
179. Parnas, "On the Criteria To Be Used in Decomposing Systems Into Modules," Communications of the ACM, Vol. 5, No. 12, December 1972, pp. 1053 - 1058.
180. Parnas, "On the Design and Development of Program Families," IEEE Transactions on Software Engineering, Vol. 2, No. 1, March 1976, pp. 1 - 9.
181. Parnas, "Designing Software for Ease of Use and Extension," IEEE Transactions on Software Engineering, Vol. 5, No. 2, March 1979, pp. 128 - 157.
182. Pfleeger, A Reuse-Oriented History of Cost Estimation Models, Contel Technology Center Technical Report CTC-TR-89-020, Chantilly, Virginia, 1989.
183. Piper and W.L. Barner, "The RAPID Center Reusable Software Components (RCSs) Certification Process," Proceedings of the Ninth Annual National Conference on Ada Technology, March 4-7, 1991, pp. 32 - 38.
184. Polster, "Reuse of Software Through Generation of Partial Systems," IEEE Transactions on Software Engineering, Vol. SE-12, No. 3, March 1986, pp. 402 - 416.
185. Prieto-Diaz and G. Arango, Editors, Domain Analysis and Software Systems Modeling, IEEE Computer Society Press, Los Alamitos, California, 1991.
186. Prieto-Diaz, "Domain Analysis for Reusability," Proceedings of COMPSAC '87, 1987, pp. 23 - 29, reprinted in IEEE Tutorial: Software Reuse: Emerging Technology, Edited by W. Tracz, IEEE Catalog No. EH0278-2, IEEE Computer Society Press, Washington, D.C., 1988, pp. 347 - 353.
187. Prieto-Diaz, "Domain Analysis: An Introduction," Software Engineering Notes, Vol. 15, No. 2, April 1990, pp. 47 - 54.
188. Raj and H.M. Levy, "A Compositional Model for Software Reuse," ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 3 - 24.
189. Ratcliffe, "Report on a Workshop on Software Reuse Held at Hereford, UK on 1,2 May 1986," Software Engineering Notes, Volume 12, No. 1, January 1987, pp. 42 - 47.

190. Rational, Inc., Large-System Development and Rational Subsystems, Document Control Number 6004, Rational, Inc., Mountain View, California, November, 1986.
191. Rauch-Hindin, "Reusable Software," *Electronic Design*, Vol. 31, No. 3, February 3, 1983, pp. 176 - 193.
192. Reilly, "Roots of Reuse," *IEEE Software*, Vol. 4, No. 1, January 1987, page 4.
193. Rine, "A Short Overview of a History of Software Maintenance: As It Pertains to Reuse," *Software Engineering Notes*, Vol. 16, No. 4, October 1991, pp. 60 - 63.
194. Rocky Mountain Institute of Software Engineering, Workshop on Software Reuse -- Participant Proceedings, October 1987, summary contained in D.T. Ross, J.B. Goodenough, and C.A. Irvine, "Software Engineering: Process, Principles, and Goals," *IEEE Computer*, Vol. 8, No. 5, May 1975, pp. 17 - 27.
195. Rothrock, "Reusable Ada Products for Information Systems Development (RAPID): Reuse - Year 2000," *Proceedings of the Seventh Washington Ada Symposium*, June 25-28, 1990, pp. 1 - 7.
196. Russell, "Experiences Implementing a Reusable Data Structure Component Taxonomy," *Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium*, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 8 - 18.
197. Schettino and C.S. Koziowski, "Facilitating Reuse in a Software Engineering Environment," *Proceedings of the Ninth Annual National Conference on Ada Technology*, March 4-7, 1991, pp. 100 - 106.
198. Schmucker, "Object Orientation," *MacWorld*, Vol. 3, No. 11, November 1986, pp. 119 - 123.
199. Shlaer and S.J. Mellor, "An Object-Oriented Approach to Domain Analysis," *Software Engineering Notes*, Vol. 14, No. 5, July 1989, pp. 66 - 77.
200. Simos, "The Domain-Oriented Software Life-Cycle: Towards and Extended Process Model for Reusability," from K.E. Sivley, "Experience and Lessons Learned in Transporting Ada Software," *Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium*, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 436 - 440.
201. Sixtensson and W. Ye, "Reuse in the Telecommunications Domain Using Object-Oriented Technology and Ada," *Proceedings of the Seventh Washington Ada Symposium*, June 25-28, 1990, pp. 231 - 239.
202. ISEC Reusability Concepts, Softech, Inc., Waltham, Massachusetts, 1985.
203. Solderitsch and T. Schreyer, "A Generative Approach to Reusing of Ada Subsystems," *Proceedings of the Seventh Washington Ada Symposium*, June 25-28, 1990, pp. 55 - 64.
204. Solderitsch, K. Wallnau, and J. Thalhamer, "Constructing Domain-Specific Ada Re-Use Libraries," *Proceedings, Seventh Annual National Conference on Ada Technology*, Atlantic City, New Jersey, March, 1989, pp. 419 - 433.

205. St. Dennis, A Guidebook for Writing Reusable Code in Ada, Honeywell, Inc., Golden Valley, Minnesota, 1986.
206. A. Standish. "An Essay on Software Reuse," IEEE Transaction on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 494 - 497.
207. Staringer, "Constructing Applications From Reusable Components," IEEE Software, Vol. 11, No. 5, September 1994, pp. 61 - 68.
208. Stark and E.W. Booth, "Using Ada to Maximize Verbatim Software Reuse," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 278 - 298.
209. Steinberg, "Commercial MIS Business Systems, Object-Oriented Development and Ada -- Can They Fit Together?," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 142 - 146.
210. Taenzer, M. Ganti, and S. Podar, "Problems in Object-Oriented Software Reuse," ECOOP '89: Proceedings of the European Conference on Object-Oriented Programming, British Computer Society Workshop Series, Cambridge University Press, Cambridge, United Kingdom, 1989, pp. 25 - 38.
211. Taenzer, M. Ganti, and S. Podar, "Object-Oriented Software Reuse: The Yoyo Problem," Journal of Object-Oriented Programming, Vol. 2, No. 3, September/October 1989, pp. 30 - 35.
212. Tarumi, K. Agusa, and Y. Ohno, "A Programming Environment Supporting Reuse of Object-Oriented Software," Proceedings of the 10th International Conference on Software Engineering, April 11-15, 1988, pp. 265 - 273.
213. Tracz, "Software Reuse: Motivators and Inhibitors," Digest of Papers COMPCON, Spring 1987, IEEE Catalog Number 87CH2409-1, Computer Society Order Number 764, Computer Society Press of the IEEE, Washington, D.C., pp. 358 - 363.
214. Tracz, "Ada Reusability Efforts: A Survey of the State of the Practice," Proceedings of the Joint Ada Conference, Fifth National Conference on Ada Technology and Washington Ada Symposium, U.S. Army Communications-Electronics Command, Fort Monmouth, New Jersey, pp. 35 - 44.
215. Tracz, "Reusability Comes of Age," IEEE Software, Vol. 4, No. 4, July 1987, pp. 6 - 8.
216. Tracz, "Software Reuse Maxims," ACM Software Engineering Notes, Vol. 13, No. 4, October 1988, pp. 28 - 31.
217. Tracz, Editor, Software Reuse: Emerging Technology, IEEE Catalog Number EH0278-2, Computer Society Order Number 846, IEEE Computer Society Press, Washington, D.C., 1988.
218. Vitaletti and E. Guerrieri, "Domain Analysis Within the ISEC RAPID Center," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 460 - 470.

219. Vogel song and J. Rothrock, "Reusable Ada Products for Information Systems Development (RAPID): Lessons Learned During Pilot Operations," Proceedings of the Eighth Annual National Conference on Ada Technology, March 5-8, 1990, pp. 452 - 459.
220. Vogel song, "Reusable Ada Packages for Information Systems Development (RAPID): An Operational Center of Excellence for Software Reuse," Proceedings of TRI-Ada '89 -- Ada Technology In Context: Application, Development, and Deployment, October 23-26, 1989, Association for Computing Machinery, New York, New York, pp. 324 - 330.
221. Wald, "Software Engineering with Reusable Parts," Digest of Papers COMPCON, Spring 1987, IEEE Catalog Number 87CH2409-1, Computer Society Order Number 764, Computer Society Press of the IEEE, Washington, D.C., pp. 353 - 356.
222. Wegner, "Varieties of Reusability," Tutorial: Software Reusability, IEEE Computer Society Press, Washington, D.C., 1987, pp. 24 - 38.
223. Weide, W.F. Ogden, and M. Sitaraman, "Recasting Algorithms to Encourage Reuse," IEEE Software, Vol. 11, No. 5, September 1994, pp. 80 - 88.
224. Welch, "Parallel Processes as Reusable Components," Ada Components: Libraries and Tools - Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Edited by S. Tafvelin, Cambridge University Press, Cambridge, U.K., pp. 86 - 99.
225. Whinery and G.H. Barber, "Analytical Approach to Software Reusability," Proceedings of the 3rd Annual National Conference on Ada Technology, March 20-21, 1985, pp. 153 - 159.
226. Wirfs-Brock and B. Wilkerson, "Variables Limit Reusability," Journal of Object-Oriented Programming, Vol. 2, No. 1, May/June 1989, pp. 34 - 40.
227. Wong, A Management Overview of Software Reuse, NBS Special Publication 500-142, U.S. Government Printing Office, Washington, D.C., September 1986.
228. Woodfield, D.W. Embley, and D.T. Scott, "Can Programmers Reuse Software?," IEEE Software, Vol. 4, No. 4, July 1987, pp. 52 - 59.

REVIEW ANALYSIS MEASURES

1. Kusumoto, S.; Matsumoto, K.; Kikuno, T.: Approaches to Improving Effectiveness of Review Activities in Technical Review Process. Proceedings of the International Software Quality Exchange, San Francisco, March 10-11, 1992.

SOFTWARE FACTORY

1. Akima, N.; Ooi, F.: Industrializing Software Development: A Japanese Approach. IEEE Software, March 1989, pp. 13-21.
2. Basili, V.R.; Caldiera, G.; Cantone, G.: A Reference Architecture for the Component Factory. ACM Transactions on Software Engineering and Methodology, 1(1992)1, pp. 53-80.
3. Cusumano, M.A.: The Software Factory: A Historical Interpretation. IEEE Software, March 1989, pp. 23-30.

4. Fernstroem, C.; Naerfelt, K.; Ohlsson, L.: Software Factory Principles, Architecture, and Experiments. IEEE Software, March 1992, pp. 36-44.

SOFTWARE MANAGEMENT

1. Ashley, N.: Measurement as a Management Tool. in Kelly, M.: Management and Measurement of Software Quality, UNICOM SEMINARS, Middlesex, UK, 1993, pp. 31-50.
2. Bazzana, G. et. al.: Software management-by-metrics: practical experiences in Italy. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29 September - 1 October 1993, Section 6.
3. Belady, L.A.; Lehman, M.M.: A model of large program development. IBM System Journal, 3 (1976), pp. 225-252.
4. Boone, G.: The Applicability of TQM to Software Development. Proceedings of the International Software Quality Conference, Dayton, Ohio, October 7-9, 1991, pp. 212-219.
5. Cardenas-Garcia, S.; Zelkowitz, M.V.: A Management Tool For Evaluation of Software Design. IEEE Transactions on Software Engineering, 17(1991)9, pp. 961-971.
6. Curtis, B.: Maintaining the Software Process. Proceedings of the Conference on Software Maintenance, Orlando, Nov. 9-12 1992, pp. 2-8.
7. Debou, C.; Liptak, T.; Pescoller, L.: Managing Software Process by Applying ami. Proceedings of the fourth IFAC/IFIP Workshop, Schloss Seggau, Austria, May 1992, pp. 103-108.
8. DeMarco, T.; Lister, T.: Peopleware (Wien wartet auf Dich! Der Faktor Mensch im DV-Management). Carl Hanser Publisher, Munich, 1991.
9. Dixon, D.: Integrated Support for Project Management. IEEE Software, 1988, pp. 49-58.
10. Doerfel, F.: Software Projects are Fuzzy (german). Proceedings of the Tool'91 Conference, Karlsruhe, November 1991, pp. 147-160.
11. Ebert, C.: Complexity Traces -- An Instrument for Software Project Management. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29 September - 1 October 1993, Section 17.
12. Fisher, D.T.: Myths and Methods - a guide to software productivity. Prentice Hall Inc., 1991.
13. Fruehauf, K.; Ludewig, J.; Sandmayr, H.: Software Project Management and Quality Assurance. (german), Teubner Publ. Stuttgart, 1993.
14. Gilb, T.: Principles of Software Engineering Management. Addison-Wesley Publ. Comp., New York, 1988.
15. Gonzalez, R.R.: A Unified Metric of Software Complexity: Measuring Productivity, Quality and Value. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon.

16. Goodman, P.: Practical Implementation on Software Metrics. McGraw-Hill Inc., 1993.
17. Humphrey, W.S.: Managing the Software Process. Addison-Wesley Publ., 1990.
18. Issendorff, von H.: The theory of organisation: a mathematical treatment of processing structures. in Mitchell, R.J.: Managing Complexity in Software Engineering. IEE Computing Series 17, London, 1990, pp. 97-112.
19. Johnson, J.R.: The Software Factory - Managing Software Development and Maintenance. QED Information Series, 1991.
20. Keller, T.W.: The Importance of Process Improvement in Software Maintenance. Tutorial, IBM Houston, 1992.
21. Kitchenham, B.A.: Software metrics and integrated project support environments. Software Engineering Journal, January 1986, pp. 58-64.
22. Kitchenham, B.A.; Walker, J.G.: A quantitative approach to monitoring software development. Software Engineering Journal, January 1989, pp. 2-14.
23. Kuntzmann-Combelles, A.: Metrics for Management. in: Fenton/Littlewood: Software Reliability and Metrics, Elsevier Applied Science, London New York, 1991, pp.~39-47.
24. Kuntzmann-Combelles, A.: Quantitative Approach to Software Management: The AMI Method. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 170-177.
25. Lanphar, R.: Quantitative Process Management in Software Engineering, A Reconciliation Between Process and Product Views. The Journal of Systems and Software, 12 (1990), pp. 243-248.
26. Liu, L.; Horowitz, E.: A Formal Model for Software Project Management. IEEE Transactions on Software Engineering, 15(1989)10, pp. 1280-1293.
27. Miluk, G.: Procee Oriented Problem Solving. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 262-274.
28. Mueller, H.A.; Corrie, B.D.; Tilley, S.R.: Spatial and Visual Representations of Software Structures - A Model for Reverse Engineering. Department of Computer Science, University of Victoria, DCS-169-IR, September 1991.
29. Neuendorf, St.: Productivity Measurement. MetricViews, January 1995, pp. 24-25.
30. Norris, M.; Rigby, P.; Payne, M.: The Healthy Software Project - A Guide to Successful Development and Management. John Wiley & Sons, 1993.
31. Paul, R.A.: Metrics to Improve the US Army Software Development Process. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 40-50.
32. Ross, N.: The Collection and Use of Data for Monitoring Software Projects. in: Kitchenham;Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 125-154.

33. Selby, R.W.: Measurement-Driven Analysis Techniques for Systematic Process Improvement. Proceedings of the International Software Quality Exchange, San Francisco, March 10-11, 1992.
34. Tate, G.; Verner, J.; Jefferey, R.: CASE: A Testbed for Modeling, Measurement and Management. Comm. of the ACM, 35(1992)4, pp. 65-72.
35. Youll, D.P.: Making Software Development Visible. John Wiley & Sons, 1990.

SOFTWARE PROCESS IMPROVEMENT

1. Ambriola, V.; Ciancarini, P.; Corradini, A.; DeFrancesco, N.: Towards Innovative Software Engineering Environments. The Journal of Systems and Software, 14(1991) pp. 17-29.
2. Arthur, L.J.: Rapid Evolutionary Development John Wiley & Sons Inc., 1992.
3. Basili, V.R.; Reiter, R.W.: An Investigation of Human Factors in Software Development. Second Edition to Selected Reprints in Software, IEEE Computer Society, Spring, 1982, p. 182-199.
4. Basili, V.R.; Rombach, H.D.: Tailoring the Software Process to Project Goals and Environments. Proceedings of the 9th International Conference on Software Engineering, Monterey, March 30 - April 2, 1987, pp. 345-357.
5. Basili, V.R.; Selby, R.W.: Calculation and Use of an Environment's Characteristics Software Metric Set. Proceedings of the 8th Conference on Software Engineering, London, August 1985, pp. 386-388.
6. Boehm, B.W.: A Spiral Model of Software Development and Enhancement. IEEE Computer, May 1988, pp. 61-72.
7. Boehm, B.W.: Software Engineering Economics. Prentice Hall, 1981.
8. Briand, L.C.; Basili, V.R.; Thomas, W.M.: A Pattern Recognition Approach for Software Engineering Data Analysis. IEEE Transactions on Software Engineering, 18(1992)11, pp. 931-942.
9. Brooks, F.P.: The Mythical Man-Month - Essays on Software Engineering. Addison Wesley Publ. Comp., 1975.
10. Budde, R.; Neumann, M.; Sylla, K.; Zuellinghoven, H.: Experiences in the object oriented design and analysis (German). GMT research paper, Sankt Augustin, (14 p.), May 1992.
11. Costello, R.; Liu, D.: Metrics for Software and Systems Requirements Engineering for Large System Acquisition and Development. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon.
12. Debou, C.: ami: a new paradigm for software process improvement. Proceedings of the International Conference on Practical Improvement of Software Processes and Products, 4-5 May, 1994 Dublin, Ireland.
13. Debou, C.; Fuchs, N.; Saria, H.: Selling Believable Technology. IEEE Software, November 1993, pp. 22-27.

14. Debou, C.; Liptak, J.; Schippers, H.: Decision making for software process improvement: a quantitative approach. Proceedings of the 2nd International Conference on Achieving Quality in Software (AQuIS 93), Venice, October 1993, pp 363-377.
15. Ebert, C.: Complexity measures in the software development process (German). Proceedings of the GI-Jahrestagung 1992, Berlin, pp. 189-198.
16. Ejiogu, L.: Software Engineering with Software Metrics. QED Information Sciences, 1991.
17. Fagan, M.E.: Inspecting Software Design and Code. Datamation, October 1977, pp. 133-144.
18. Franke, D.W.; Purvis, M.K.: Hardware/Software CoDesign: A Perspective. Proceedings of the IEEE, 1991, pp. 344-352.
19. Johnson, M.A.: Experience with Quality Metrics for Tracking Software Development. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon.
20. Karam, G.M.; Casselman, R.S.: A Cataloging Framework for Software Development Methods. IEEE Computer, February 1993, pp. 34-46.
21. Karunanithi, N.; Malaiya, Y.K.: The Scaling Problem in Neural Networks for Software Reliability Prediction. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, pp. 76-82.
22. Kitchenham, B.A.; Littlewood, B.: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989.
23. Khoshgoftaar, T.M.; Lanning, D.L.: An Neural Network Approach for Early Detection of Program Modules having High Risk in the Maintenance Phase Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon.
24. Kugler, H.: Process Improvement: How Much Can the Organisation Endure? Proceedings of the International Conference on Practical Improvement of Software Processes and Products, 4-5 May, 1994, Dublin, Ireland.
25. Lawrence, M.J.: Programming Methodology, Organizational Environment, and Programming Productivity. The Journal of Systems and Software, 2 (1981), pp. 257-269.
26. Lee, I.; Iyer, R.K.: Analysis of Software Halts in the Tandem GUARDIAN Operating System. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, pp. 227-236.
27. Lott, C.M.: Process and measurement support in SEEs. Software Engineering Notes, 18(1993)4, pp. 83-93.
28. Mayrhauser, A.: Software Engineering. Academic Press, Inc. Boston New York, 1990.
29. Messnarz, R.; Kugler, H.; Haase, V.: BOTTSTRAP and ISO 9000: A Quantitative Approach to Objective Quality Management. Proceedings of the International Conference on Practical Improvement of Software Processes and Products, 4-5 May, 1994, Dublin, Ireland.

30. Mills, H.D.; Basili, V.R.; Gannon, J.D.; Hamlet, R.G.: Mathematical Principles for a First Course in Software Engineering. IEEE Transactions on Software Engineering, 15(1989)5, pp. 550-559.
31. Mills, H.D.; Dyson, P.B.: Using Metrics Quantify Development. IEEE Software, March 1990, pp. 14-16.
32. Moeller, K.-H.: Metric application in the software development. (German). HMD, (1992) 163, pp. 17-30.
33. Mitchell, R.J.: Managing Complexity in Software Engineering. IEEE Computing Series 17, London, 1990.
34. Morell, L.; Murrill, B.: Semantic Metrics through Error Flow Analysis. The Journal of Systems and Software, 20(1993)3, pp. 253-265.
35. Myers W.: A Statistical Approach to Scheduling Software Development. Second Edition to Selected Reprints in Software, IEEE Computer Society, Spring, 1982, pp. 105-117.
36. Natale, D.: On the Impact Metrics' Application in a Large Scale Software Maintenance Environment. Proceedings of the Conference on Software Maintenance 1991, Sorrento, Italy, October 15-17, pp. 114-118.
37. Olsen, N.C.: The Software Rush Hour. IEEE Software, September 1993, pp. 29-37.
38. Paulish, D.J.; Carleton, A.D.: Case Studies of Software-Process-Improvement Measurement. IEEE Computer, September 1994, pp. 50-57.
39. Paulk, M.C.: The Capability Maturity Model: A Tutorial. Tutorial on the Second International Conference on Software Quality, Research Triangle Park, NC, October 1992.
40. Pfleeger, S.L.; Rombach, H.D.: Measurement Based Process Improvement. IEEE Software, July 1994, pp. 9-11.
41. Pressman, R.S.: Software Engineering - A Practitioner's Approach. McGraw-Hill Int., 1987.
42. Putnam, L.H.: Measures for Excellence. Unicom Publisher, 1992.
43. Ramamoorthy, C.V.; Garg, V.; Prakash, A.: Programming in the Large. IEEE Transactions on Software Engineering, 12(1986)7, pp. 769-783.
44. Ramamoorthy, C.V.; Prakash, A.; Tsai, W.; Usuda, Y.: Software Engineering: Problems and Perspectives. IEEE Computer, Oct. 1984, pp. 191-209.
45. Rubin, H.A.: Software Process Maturity: Measuring its Impact on Productivity and Quality. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 2-10.
46. Rudisill, D.: QCASE - A New Paradigm For Computer Aided Software Engineering. Proceedings of the International Software Quality Exchange, San Francisco, March 10-11, 1992.

47. Selby, R.W.: Interconnectivity Analysis Techniques for Error Localization in Large Systems. *The Journal of Systems and Software*, 20(1993)3, pp. 267-294.
48. Shepperd, M.: *Software Engineering Metrics*. Vol. I and II, McGraw-Hill, 1993/94.
49. Shneiderman, B.: *Software Psychology*. Winthrop Publishers Inc., Massachusetts, 1980.
50. Slovin, M.; Nunno, D.D.: Engineering the Enterprise: Envolving Quality Systems. *ACM SIGSOFT Notes*, July 1994, pp. 45-47.
51. Stahlknecht, P.: The application of mathematics to the software development (German). *Angewandte Informatik*, 24(1982)2, pp. 115-125.
52. Tsui, F.; Hofmann, S.C.; Goldstrohm, W.J.: A software development post-mortem summary. *Software Engineering Journal*, July 1992, pp. 277-284.
53. Wilkinson, S.J.: Deploying Software Metrics in a Large Organisation. *Proceedings of the Annual Oregon Workshop on Software Metrics*, April 10-12, Silver Falls, Oregon.

SOFTWARE QUALITY

1. Agresti, W.W.; Evanco, W.M.: Projecting Software Defects From Analyzing Ada Designs. *IEEE Transactions on Software Engineering*, 18(1992)11, pp. 988-997.
2. Arthur, L.J.: *Improving Software Quality - An Insider's Guide to TQM*. John Wiley & Sons, 1993.
3. Asam, R.; Drenkard, N.; Maier, H.: *Quality evaluation of software products* (German). Berlin, Munich, 1986.
4. Azuma, M.: *Evaluation of Software Quality*. INSTAC Research Report, Tokio, 1991.
5. Bernstein, L.: *Notes on Software Quality Management*. *Proceedings of the International Software Quality Exchange*, San Francisco, March 10-11, 1992.
6. Boegh, J.: *SCOPE: A Guide for Software Product Quality Evaluation*. *Proceedings of the International Conference on Practical Improvement of Software Processes and Products*, 4-5 May, Dublin, Ireland.
7. Card, D.: Beyond quality to customer satisfaction. *IEEE Software*, March 1992, pp. 101-102.
8. Carey, D.R.: *Quality Measurements in Software*. *Proceedings of the International Software Quality Conference*, Dayton, Ohio, October 7-9, 1991, pp. 19-24.
9. Cobb, R.H.; Mills, H.D.: *Engineering Software under Statistical Quality Control*. *IEEE Software*, November 1990, pp. 44-54.
10. Compton, B.T.; Withrow, C.: Prediction and Control of ADA Software Defects. *The Journal of Systems and Software*, 12 (1990), pp. 199-207.
11. Conte, S.D.; Dunsmore, H.E.; Shen, V.Y.: *Software Engineering Metrics and Models*. The Benjamin/Cummings Publ. Comp. Inc., 1986.

12. Delen, G.P.; Rijsenbrij, D.B.: The Specification, Engineering, and Measurement of Information Systems Quality. *The Journal of Systems and Software*, 17(1992), pp. 205-217.
13. Denning, P.J.: What is software quality? *Comm. of the ACM*, 35(1992)1, pp. 13-15.
14. Dippold, U.: A Framework of Metrics for Software Quality Assurance in all Software Development Phases. Master's Thesis, University of Erlangen, Germany, 1992.
15. Doolan, E.P.: Expierence with Fagan's Inspection Method. *Software -- Practice and Experience*, 22(1992)2, pp. 173-182.
16. Edgar-Nevill, V.; Edgar-Nevill, D.: Measuring software quality improvement: a case study. *Proc. of the Fourth European Conference on Software Quality*, October 17-20, Basel, Switzerland, pp. 400-411.
17. Ehrenberger, W.: What is reliable software ? (German). *Elektronische Rechenanlagen*, Munich, 25(1983)1, pp. 27-32.
18. Evans, M.W.; Marchiniak, J.J.: *Software Quality Assurance and Management*. John Wiley & Sons, 1987.
19. Ey, W.; Schumacher, H.: Metric programs are one basis for software quality (German). *Computerwoche*, 6(1992) February, pp. 13-16.
20. Fagan, M.E.: Advances in Sofwtare Inspections. *IEEE Transactions on Software Engineering*, 12(1986)7, pp. 744-751.
21. Fagan, M.E.: Design and code inspections to reduce errors in program development. *IBM System Journal*, 15(1976)3, pp. 182-211.
22. Fruehauf, K.: Limits of the software quality management (German). in: *Wirtschaftlichkeit von Software-Entwicklung und -Einsatz*, Teubner Publisher, Stuttgart, German Chapter of ACM Berichte 36, 1992, pp. 257-268.
23. Fruehauf, K.; Ludewig, J.; Sandmayr, H.: *Software Examination* (German). Teubner Publisher, Stuttgart, 1991.
24. Goetzke, C.: Principles and methods of the software quality assurance in a concrete development environment (German). Study, Siemens Nixdorf Munich, Technical University of Magdeburg, 1993.
25. Grady, R.B.: Dissecting Software Failures. *Hewlett-Packard Journal*, 40(1989)2, pp. 57-63.
26. Grady, R.B.: Practical Results From Measuring Software Quality. *Comm. of the ACM*, 36(1993)11, pp. 62-68.
27. Grady, R.B.: The Role of Software Metrics in Managing Quality and Testing. *Hewlett-Packard Report*, 1/30/90 pp. 1-15.
28. Grady, R.B.: Work-Product Analysis: The Philosopher's Stone of Software? *IEEE Software*, March 1990, pp. 26-34.

29. Hatton, L.: A case history of automated incremental improvement of software product quality. Proceedings of the Tenth Annual Conference of Software Metrics and Quality Assurance in Industry, Amsterdam, 29 September - 1 October 1993, Section 27.
30. Hausen, H.L.: A Rule-Based Approach to Software Quality Engineering. in: Fenton/Littlewood: Software Reliability and Metrics, Elsevier Applied Science, London New York, 1991, pp. 48-68.
31. Hausen, H.L.: Generic Modeling of Software Quality. in: Kitchenham;Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 201-242.
32. Hausen, H.L.; Muellerburg, M.; Schmidt, M.: Examination, Measurement and Assessment of Software Products and Projects. Proceedings of the 1st International Congress of the EOQC, June 1987.
33. Hayes, B.E.: Measuring Customer Satisfaction -- Development and Use of Questionnaires. ASQC Quality Press, Milwaukee, Wisconsin, 1992.
34. Hirayama, M.; Sato, H.; Yamada, A.; Tsuda, J.: Practice of quality modeling and measurement on software life-cycle. Proceedings of the 12th International Conference on Software Engineering, March~26-30, Nice, France, 1990, pp. 98-107.
35. Hoecker, H.; Itzfeldt, W.D.; Schmidt, M.; Timm, M.: Comparative Descriptions of Software Quality Measures. GMD Study No. 81, Birlinghoven, 1984.
36. Itzfeldt, W.D.: Quality metrics for software management and engineering. in Mitchell, R.J.: Managing Complexity in Software Engineering. IEE Computing Series 17, London, 1990, pp. 127-152.
37. Itzfeldt, W.D.; Timm, M.: A Systematic of Description for Quality Measures (german). Angewandte Informatik, 25(1983)7, pp. 273-281.
38. Jannasch, H.: Investigations to software metrics for their applicability to the evaluation of software quality (german). Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, October 1990, pp. 43-49.
39. Joh, F.: Enabling Software Quality Through Total Quality Management, Process Improvement, and Technology Insertion. Proceedings of the International Software Quality Conference, Dayton, Ohio, October 7-9, 1991, pp. 152-157.
40. Keller, S.E.; Kahn, L.G.; Panara, R.B.: Specifying Software Quality Requirements with Metrics. in Thayer, R.H.; Dorfman. M.: System and Software Requirements Engineering, IEEE Computer Society Press, Washington, 1990, pp. 145-163.
41. Kelly, M.: The Management and Measurement of Software Quality. Unicom Publisher, 1992.
42. Khoshgoftaar, T.M.; Pandya, A.S.; More, H.B.: A Neural Network Approach for Predicting Software Development Faults. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, pp. 83-89.
43. Kitchenham, B.A.: Software quality assurance. Microprocessors and Microsystems, 13(1989)6, pp. 373-381.

44. Kitchenham, B.A.: Towards a constructive quality model. *Software Engineering Journal*, July 1987, pp. 105-113.
45. Kuehnel, B.: Practice of an Integrated Software Quality Assurance (german). *Softwaretechnik-Trends*, 13(1993)1, Germany, pp. 35-42.
46. Kuhrau, I.: Software quality assurance (german). Study, CAP debis Curadata, Technical University of Magdeburg, 1993.
47. Luong, D.: Software Development Process Quality Capability Modelling for Continuous Improvement. *Proceedings of the Second International Conference on Software Quality*, Research Triangle Park, NC, October 5-7, pp. 70-77.
48. Luise, R.: The Evaluation of Source Program Quality Metrics. *Proceedings of the Second European Conference on Software Quality Assurance*, Oslo, 1990.
49. Markfort, D.: Japanese Methods of Quality Engineering (german). *CIM*, 5/92, pp. 15-24.
50. MHB: Manual of methods for system software development (german). SNI, Paderborn, September 1990.
51. Moeller, K.-H.: New trends in software quality assurance in the point of view at users (german). *Softwaretechnik-Trends*, 7-1 (1987), pp. 26-39.
52. Moeller, K.-H.; Paulish, D.J.: Software metrics in practice (german). Oldenbourg Publisher, Munich Wien, 1993.
53. Nejme, B.A.: Software Quality: Myths, Malpractices and Mandates. *Proceedings of the International Software Quality Exchange*, San Francisco, March 10-11, 1992.
54. Nivoix, J.: SQA in a changing environment. *Proceedings of the Workshop Computer Aided Software Evaluation*, TU Magdeburg, October 1990.
55. Ortiz, E.A.: Predicting Software Quality: The Generalized Goel-Okumoto Model. *Proceedings of the Second International Conference on Software Quality*, Research Triangle Park, NC, October 5-7, pp. 148-154.
56. Perry, W.E.: *Quality Assurance for Information Systems*. QED Technical Publishing Group, 1991.
57. Redmill, F.J.: Considering quality in the management of software-based development projects. *Information and Software Technology*, 32(1990)1, pp. 18-22.
58. Robillard, P.N.: *Le Logiciel: de sa conception a sa maintenance*. Gaetan morin editeur, Quebec, 1985.
59. Rombach, H.D.: Quantitative evaluation of software quality characteristics on the base of structurally measures (german). Ph. Dissertation, University of Kaiserslautern, 1984.
60. Schaefer, H.: Making Software Quality Assurance Happen Experiences With the Introduction of SQA. *Proceedings of the International Software Quality Exchange*, San Francisco, March 10-11, 1992.

61. Scheibl, H.-J.: Commercial software development (Kommerzielle Software-Entwicklung). expert Publisher, Boeblingen, 1989.
62. Schneidewind, N.F.: Controlling and Predicting the Quality of Space Shuttle Software sing Metrics. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon.
63. Schnurer, K.E.: Program Inspections -- Experiences and Problems (german). Informatik-Spektrum, 11(1988), pp. 312-322.
64. Shaughnessy, E.P.: Measuring the Quality of Software. Proceedings of the International Software Quality Conference, Dayton, Ohio, 1991, pp. 38-42.
65. Simmons, A.: Interfacing Numbers with People: Lessons in Quality Maesurement Techniques. IEEE Communication Magazin, October 1994, pp.36-40.
66. Suresh, N.; Babu, A.J.G.: Software Development: Enhancing Quality and Cost-Effectiveness. Proceedings of the Third International Confenrence on Software Quality, Lake Tahoe, Nevada, 4-6 October 1993, pp. 85-92.
67. Tajima D.; Matsubara, T.: The Computer Software Industry in Japan. IEEE Computer, May 1991, pp. 89-96.
68. Talley, D.J.: Total Quality Management -- Performance and Cost Measures. ASQC Quality Press, Milwaukee, Wisconsin, 1991.
69. Trammell, C.J.; Poore, J.H.: A Group Process for Defining Local Software Quality: Field Applications and Validation Experiments. Software -- Practice and Experience, 22(1992)8, pp. 603-636.
70. Valette, V.; Vallee, F.: Software quality metrics in space systms. Proceedings of the Thrid International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, pp. 296-302.
71. Voss, A.J.R.: The need for a quality culture for software development. in Mitchell, R.J.: Managing Complexity in Software Engineering. IEEE Computing Series 17, London, 1990, pp. 113-126.
72. Vincent, J.; Waters, A.; Sinclair, J.: Software Quality Assurance. Vol. I u. II, Prentice Hall, New Jersey, 1988.
73. Walker, J.G.; Kitchenham, B.A.: Quality Requirements Specification and Evaluation. in: Kitchenham;Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 103-124.
74. Wallmueller, E.: Application and Experiences with Software Metrics. Proceedings of the Second European Conference on Software Quality Assurance, Oslo, 1990.
75. Wallmueller, E.: Quality Management and Metrics. in Kelly, M.: Management and Measurement od Sofwtare Quality, UNICOM SEMINARS, Middlesex, UK, 1993, pp. 65-78.
76. Wallmueller, E.: Software quality assurance in practice (german). Carl Hanser Publisher, Munich Wien, 1990.

77. Wallmueller, E.: Software quality assurance through the information engineering (german). in: Wirtschaftlichkeit von Software-Entwicklung und -Einsatz, Teubner Publisher, Stuttgart, German Chapter of ACM Berichte 36, 1992, pp. 7-25.
78. Wallmueller, E.: Software quality assurance in a industrial environment (german). Softwaretechnik-Trends, 7-1 (1987), pp. 40-49.
79. Weerahandi, S.; Hausman, R.E.: Software Quality Measurement Based on Fault-Detection Data. IEEE Transactions on Software Engineering, 20(1994)9, pp. 665-676.
80. Weinberg, G.M.: Quality Software Management. Dorset House Publishing, New York, 1993.

SOFTWARE QUALITY ASSURANCE

1. Bache, R.; Leelasena, L.: QUALMS -- A Tool for Control Flow Analysis and Measurement. South Bank Polytechnic, London, (9 p.) May 1991.
2. Foltin, E.; Pilz, S.: A tool to the computer based analysis of the topology in real networks with the use of PC-METRIC (german). Master's Thesis, TU Magdeburg, 1993.
3. Fontaine; Neel,D.; Segot, J.: Computer-Based Systems Quality: The MACSI Method. Second European Conference on Software Quality Assurance, Conf.Proc., Oslo, 1990.
4. Gustafson, G.G.; Kerr, R.J.: Some Practical Experience with a Software Quality Assurance Program. Comm. of the ACM, 25(1982)1, pp. 4-12.
5. Hoffmann, B.: Application and Evaluation of a Computer Aided Measurement Tool for the Evaluation of the Project Quality (german). Study, TU Magdeburg, June 1989.
6. Honda, N.; Mano, T.; Hirai, Y.: Quality Assurance System throughout Software Life Cycle. Second European Conference on Software Quality Assurance, Conf.Proc., Oslo, 1990.
7. Humblot, D.: Manuel Qualit`e des Etudes. R`eseaux et syst`emes d'information, Bull, Paris 1990.
8. Kan, S.H.: Applying the Seven Basic Quality Tools in Software Development. Proceedings of the International Software Quality Exchange, San Francisco, March 10-11, 1992.
9. Kuehne, S.: Integration of SQA Tools in CASE Tools (german). Study, TU Magdeburg, July 1991.
10. Leiste, H.: Conception of a Software Evaluation Tool (Konzeption eines Softwarebewertungsplatzes). Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, October 1990, pp. 50-62.
11. Leiste, H.: Implementation of a prototype of a software evaluation tool (Implementation des Prototyps eines Softwarebewertungsplatzes). Master's Thesis, TU Magdeburg, 1991.
12. LOGISCOPE: Part of the Software Quality Report. VERILOG, Munich, 1991.
13. LOGISCOPE: Editor 3.1, User's Manual, Verilog, Toulouse, Dec. 1991.

14. LOGISCOPE: Analyzers, Reference Manual, Verilog, Toulouse, Febr. 1992.
15. Mathes, W.: CASE and software quality assurance (german). Proceedings of the TOOL'91, Karlsruhe, November 1991, pp. 291-296.
16. Mueller, S.: Investigations to metric based CASE tools (german). Master's Thesis, TU Magdeburg, 1992.
17. QUALIGRAPH - An Automated Tool for Software Quality Control & Graphic Documentation - General Information Manual, Szki, Budapest, 1989.
18. Sneed, H.M.; Merey, A.: Automated Software Quality Assurance. IEEE Transactions on Software Engineering, 11(1983)9, pp.~909-916.
19. Szentes, J.; Jannasch, H.: QUALIGRAPH -- a tool to the measurement of software quality and the graphical presentation of software (german). Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, October 1990, pp. 80-82.
20. Tiedge, I.: Extension of the software evaluation program SVS to a measurement program (german). Master's Thesis, TU Magdeburg, 1992.
21. Tsalidis, C.: {Software Quality Measurement Tools. Athens Technology Center, Athen, 1991.
22. Yamada, A.; Hirayama, M.; Sato, H.; Tsuda, J.: Quantitative Analysis Method of Software Design Characteristics for Quality Improvement. Second European Conference on Software Quality Assurance, Conf. Proc., Oslo, 1990.

SOFTWARE SYSTEMS/TREE STRUCTURE

1. Balzert, H.: Quantitative methods to the determination of the complexity of software systems (german). Lecture Notes on Computer Science 50, Springer Publisher, Berlin Heidelberg New York, 1981.
2. Blaschek, G.: Static program analysis (german). Elektronische Rechenanlagen, Munich, 27(1985)2, pp. 89-94.
3. Card, D.N.; Glass, R.L.: Measuring Software Design Quality. Prentice-Hall Inc., New Jersey, 1990.
4. Dumke, R.: Software Metrics in the Software Design. Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, October 1990, pp. 30-42.
5. Ebert, C.: Integrated Automated Software Design Measures: Results from a CASE-Study. Proceedings of the 10th Annual Software Reliability Symposium, Denver, June 25-26, 1992, pp. 93-100.
6. Ebert, C.; Riegg, A.: A Framework for Selecting System Design Metrics. Proceedings of the 10th Annual Software Reliability Symposium, USA, 1991, pp. 12-19.
7. Fenton, N.E.; Kaposi, A.A.: An Engineering Theory of Structure and Measurement. in: Kitchenham; Littlewood: Measurement for Software Control and Assurance. Elsevier Science Publisher Ltd, 1989, pp. 289-320.

8. Khoshgoftaar, T.M.; Munson, J.C.; Lanning, D.L.: Dynamic System Complexity. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 129-140.
9. Harrison, W.; Cook, C.R.: A Micro/Macro Measure of Software Complexity. The Journal of Systems and Software, 7 (1987), pp. 213-219.
10. Martin, D.; Estrin, G.: Models of Computations and Systems - Evaluation of Vertex Probabilities in Graph Models of Computations. Journal of the ACM, 1967.
11. Mueller, H.A.: Verifying Software Quality Criteria Using an Interactive Graph Editor. Dept. of Computer Science, University of Victoria, Canada, DCS-139-IR, August 1990.
12. Mueller, H.A.; Uhl, J.S.: Composing Subsystem Structures Using (k,2)-partite Graphs. Proceedings of the Conference on Software Maintenance, San Diego, November 28-29, 1990, pp. 12-19.
13. Porter, A. A.; Selby, R. W.: Empirically Guided Software Development Using Metric-Based Classification Trees. IEEE Software, March 1990, pp. 46-54.
14. Rice, M.D.: Complexity Measures on Trees. in: Denvir et. al.: Formal Aspects of Measurement. Springer Verlag, 1992, pp. 108-115.
15. Riegg, A.: A method for computer aided use of metrics in the design of automation systems (german). Ph. dissertation, University of Stuttgart, 1992.
16. Rombach, H.D.: Design Measurement: Some Lessons Learned. IEEE Software, March 1990, pp. 17-25.
17. Sjoberg, D.: Quantifying schema evolution. Journal of Information and Software Technology, 35(1993)1, pp. 35-44.
18. Yamada, A.; Hirayama, M.; Sato, H.; Tsuda, J.: Quantitative Analysis Method of Software Design Characteristics for Quality Improvement. Second European Conference on Software Quality Assurance, Conf.Proc., Oslo, 1990.

SOURCE CODE MEASURES

1. Aggarnal, K.K.; Singh, Y.: A Modified Form of Software Science Measures. ACM SIGSOFT Notes, 19(1994)3, pp. 48-50.
2. Baecker, R.M.; Marcus, A.: Human Factors and Typography for More Readable Programs. Addison-Wesley Publ. Comp., 1991.
3. Benander, A.C.; Benander, B.A.: An Empirical Study of COBOL Programs Via a Style Analyzer: The Benefits of Good Programming Style. The Journal of Systems and Software, 10(1989)4, pp. 271-279.
4. Bernard, J.; Price, A.: Managing Code Inspection Information. IEEE Software, March 1994, pp. 59-69.
5. Berns, G.M.: Assessing Software Maintainability. Comm. of the ACM, 27(1984)1, pp. 14-23.

6. Brauns, E.: Compared application of software metrics to programming languages Pascal, Fortran IV, Smalltalk and Prolog (german). Master's Thesis, TU Berlin/TU Magdeburg, 1991.
7. Harrison, W.; Cook, C.: Are Deeply Nested Conditionals Less Readable? The Journal of Systems and Software (1986)6, pp. 335-341.
8. Huber, A.R.: Experience with Code Reviews in DG/UX. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 8-14.
9. Moeller, K.; Paulish, D.J.: An Empirical Investigation of Software Fault Distribution. Proceedings of the First International Software Metrics Symposium, Baltimore, May 21-22, 1993, pp. 82-90.
10. Taesley, B.E.: The effects of naming style and expertise on program comprehension. International Journal of Human-Computer Studies, (1994)40, pp. 757-770.
11. Triantafyllos, G.; Vassiliadis, S.; Kobrosly, W.: Software Metrics for the Microcode of Computer Systems. The Journal of Systems and Software, 26(1994)3, pp. 221-232.
12. Zelnowski, J.; Simmons, D.B.: Measuring program complexity in a COBOL environment. Proceedings of the AFIPS Conference, Vol. 49, 1980, pp. 757-76.

STANDARDS

1. Bons, H.; Salmann, S.: Software Quality Assurance and Software Standards (german). Wirtschaftsinformatik, 34(1992)4, pp. 401-412.
2. Bowen, J.B.: A Survey of Standards and Proposed Metrics for Software Quality Testing. Second Edition to Selected Reprints in Software, IEEE Computer Society, Spring, 1982, pp. 176-181.
3. DIN 55350: Notions on software quality assurance (german). Beuth Publisher, Berlin, 1987.
4. DIN/ISO 9000 (EN 29000): Software quality management (german). Beuth Publisher, Berlin, 1990.
5. DIN/ISO 9001 (EN 29001): Systems for software quality assurance I (german). Beuth Publisher, Berlin, 1990.
6. DIN/ISO 9002 (EN 29002): Systems for software quality assurance II (german). Beuth Publisher, Berlin, 1990.
7. DIN/ISO 9003 (EN 29003): Systems for software quality assurance III (german). Beuth Publisher, Berlin, 1990.
8. DIN/ISO 9004 (EN 29004): Software quality management and the elements of a system for software quality assurance (german). Beuth Publisher, Berlin, 1990.
9. Galinski, C.: Standardization activities. Computer Standards & Interfaces, 14(1992)4, pp. 339-353.

10. IEEE Guide for Software Quality Assurance Planning. IEEE Std 983-1986, New York, January 1986.
11. IEEE Guide for the Use of IEEE Standard Dictionary of Measures to Produce Reliable Software. IEEE Std 982.2-1988, New York, June 1989.
12. IEEE Standard Dictionary of Measures to Produce Reliable Software. IEEE Std 982.1-1988, New York, April 1989.
13. IEEE Standard for a Software Quality Metrics Methodology. IEEE Std 1061-1992, New York, March 1993.
14. ISO/IEC 9126 Standard for Information Technology, Software Product Evaluation -Quality Characteristics and Guidelines for their use. Geneve 1991.
15. Sanz, J.G.: Software QA/QM Standards for the 90's. Proceedings of the Second International Conference on Software Quality, Research Triangle Park, NC, October 5-7, pp. 23-32.
16. Schaefer, H.: International Standard for Software Quality Assurance. Proceedings of the Workshop Computer Aided Software Evaluation, TU Magdeburg, October 1990, pp. 64-73.
17. Schneidewind, N.F.: Report on the IEEE Standard for a Software Quality Metrics Methodology. Software Engineering Notes, 18(1993)3, pp. A-95 - A-98.
18. Shepperd, M.: Early life-cycle metrics and software quality models. Information and Software Technology, 32(1990)4, pp. 311-316.
19. Siegel, S.: Why we need checks and balances to assure quality. IEEE Software, January 1992, pp. 102-103.
20. Voldner, P. : An Overview of ISO/ IEC JTC1/ SC7 Software Engineering Standards Activities. Proceedings of the Quality Engineering Workshop, Bell Canada, October 15-17, 1991, pp. 23.1-23.7.

Testing Bibliography

1. Chin-Kuei, An Introduction to Software Quality Control, John Wiley & Sons, New York, New York, 1980.
2. Chow, Editor, IEEE Tutorial: Software Quality Assurance: A Practical Approach, IEEE Computer Society Press, Silver Spring, Maryland, 1985.
3. Chusho, "Test Data Selection and Quality Estimation Based on the Concept of Essential Branches for Path Testing," IEEE Transactions on Software Engineering, Vol. SE-13, No. 5, May 1987, pp. 509 - 517.
4. Cohen, Ada As a Second Language, McGraw-Hill, New York, New York, 1986.
5. Crosby, Quality is Free, The New American Library, Inc., New York, New York, 1979.
6. DeMillo, W. McCracken, R. Martin, J. Passafiume, Software Test and Evaluation, Benjamin/Cummings, Menlo Park, California, 1987.

7. Dencker, "The Ada-IDAS Project Monitoring Task States," Proceedings of the Sixth Washington Ada Symposium, Association for Computing Machinery, New York, New York, 1989, pp. 81 - 87.
8. Deutsch, Software Verification and Validation, Prentice Hall, Englewood Cliffs, New Jersey, 1982.
9. Dijkstra, "Structured Programming," reprinted in Software Engineering, Concepts and Techniques, Buxton, Naur, Randall, Editors, Litton Educational Publishing, New York, New York, 1976.
10. Dijkstra, "Guarded Commands, Nondeterminacy, and Formal Derivations of Programs," Communications of the ACM, Vol. 18, No. 8, August 1975, pp. 453-457.
11. Dillion, R.A. Kermmerer, and L.J. Harrison, "An Experience with Two Symbolic Execution-Based Approaches to Formal Verification of Ada Tasking Programs," Proceedings of the Second Workshop on Software Testing, Verification, and Analysis, 19 - 21 July 1988, Banff, Canada, IEEE Catalog Number 88TH0225-3, 1988, pp. 114 - 122.
12. Doong and P. Frankl, "Case Studies in Testing Object-Oriented Software," Testing, Analysis, and Verification Symposium, Association for Computing Machinery, New York, New York, 1991, pp. 165 - 177.
13. R Dunn and R Ullman, Quality Assurance For Computer Software, McGraw-Hill, New York, New York, 1982.
14. Dunn, Software Defect Removal, McGraw-Hill, New York, New York, 1984.
15. Dunn, Software Quality: Concepts and Plans, Prentice Hall, Englewood Cliffs, New Jersey, 1990.
16. Evangelist, "An Analysis of Control Flow Complexity," Proceedings of the Eighth International Computer Software and Applications Conference, Chicago, Illinois, November 7-9, 1984, pp. 235 - 237, 239.
17. Evans and J. Marciniak, Software Quality Assurance and Management, John Wiley & Sons, New York, New York, 1987.
18. Evans, Productive Software Test Management, John Wiley & Sons, New York, New York, 1984.
19. Fagan, "Design and Code Inspections To Reduce Errors in Program Development," IBM Systems Journal, Vol. 15, No. 3, 1976, pp. 219 - 248.
20. Fagan, "Advances in Software Inspections," IEEE Transactions on Software Engineering, Vol. 12, No. 7, July 1986, pp. 744 - 751.
21. Fairley, Software Engineering Concepts, McGraw-Hill, New York, New York, 1985.
22. Fiedler, "Object-Oriented Unit Testing," Hewlett-Packard Journal, Vol. 36, No. 4, April 1989, pp. 69 - 74.
23. Frankl and R.K. Doong, "Tools for Testing Object-Oriented Programs," Proceedings of the Pacific Northwest Conference on Software Quality, 1990, pp. 309 - 324.

24. Frankel and E.J. Weyuker, "Data Flow Testing in the Presence of Unexecutable Paths," Proceedings of the Workshop on Software Testing, Banff, Canada, July 1987, pp. 4 - 13.
25. Frankel and E.J. Weyuker, "An Applicable Family of Data Flow Testing Criteria, IEEE Transactions on Software Engineering, Vol. 14, No. 10, October 1988, pp. 1483 - 1498.
26. Freedman and G.M. Weinberg, Handbook of Walkthroughs, Inspections, And Technical Reviews: Evaluating Programs, Projects and Products, Third Edition, Dorset House Publishing, New York, New York, 1990.
27. Freedman, "Testability of Software Systems," IEEE Transactions on Software Engineering, Vol. 17, No. 6, June 1991, pp. 553 - 564.
28. Fujiwara, G.V. Bochmann, F. Khendek, M. Amalou, and A. Ghedamsi, "Test Selection Based on Finite State Models," IEEE Transactions on Software Engineering, Vol. 17, No. 6, June 1991, pp. 591 - 603.
29. Fujiwara, Logic Testing and Design for Testability, MIT Press, Cambridge, Massachusetts, 1985.
30. Gallimore, D. Coleman, and V. Stravridou, Computer Journal, "UMIST OBJ: A Language for Executable Program Specifications," Computer Journal, Vol. 32, No. 5, October 1989, pp. 413 - 421.
31. Garfield and B. Gillian, Structured Systems Development Techniques : Strategic Planning to System Testing, Pitman, 1982.
32. Geller, "Test Data As an Aid In Proving Program Correctness," Communications of the ACM, Vol. 21, No. 5, May 1978, pp. 368 - 375.
33. Gelperin and B. Hetzel, "The Growth of Software Testing," Communications of the ACM, Vol. 31, No. 6, June 1988, pp. 687 - 695.
34. German, D.P. Helmbold, and D.C. Luckham, "Monitoring for Deadlocks in Ada Tasking," ACM 0-89791-0877/82/010/0010, Proceedings of the AdaTEC Conference on Ada, Virginia, October, 1982, pp. 10 - 25.
35. S.M German, "Monitoring for Deadlock and Blocking in Ada Tasking," IEEE Transactions on Software Engineering, Vol. SE-10, No. 6, November 1984, pp. 764 - 777.
36. Ghezzi, M. Jazayeri, and D. Mandrioli, Fundamentals of Software Engineering, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
37. Gilb, Software Metrics, Winthrop Publishers, Inc., Cambridge, Massachusetts, 1977.
38. Glass, Software Reliability Guidebook, Prentice Hall, Englewood Cliffs, New Jersey, 1979.
39. Goguen and J. Meseguer, "Unifying Functional, Object-Oriented and Relational Programming With Logical Semantics," in Research Directions in Object-Oriented Programming, Edited by B. Shriver and P. Wegner, The MIT Press, Cambridge, Massachusetts, 1987, pp. 417 - 477.

40. Goguen, "Parameterized Programming," IEEE Transactions on Software Engineering, Vol. SE-10, No. 5, September 1984, pp. 528 - 543.
41. Goodenough and S.L. Gerhart, "Toward a Theory of Test Data Selection," IEEE Transactions on Software Engineering, Vol. SE-1, June 1975, pp. 156 - 173.
42. Gould and P. Drongowski, A Controlled Psychological Study of Computer Program Debugging, Technical Report RC-4083, IBM Research Division, Yorktown Heights, New York, 1972.
43. Gould, "Some Psychological Evidence on How People Debug Computer Programs," International Journal of Man-Machine Studies, Vol. 7, No. 2, 1975, pp. 151 - 182.
44. Gourlay, "A Mathematical Framework for the Investigation of Testing," IEEE Transactions on Software Engineering, Vol. SE-9, No. 6, November 1983, pp. 686 - 709.
45. Grogono and A. Bennett, "Polymorphism and Type Checking in Object-Oriented Languages," SIGPLAN Notices, Vol. 24, No. 11, November 1989, pp. 109 - 115.
46. Gunther, Management Methodology For Software Product Engineering, John Wiley & Sons, New York, New York, 1978.
47. Halstead, Elements of Software Science, Operating and Programming Systems Series, Elsevier Computer Science Library, New York, New York, 1977.
48. Hansen, "Concurrent Programming Concepts," ACM Computing Surveys, Vol. 5, No. 4, December 1973, pp. 223 - 245.
49. Hansen, "Distributed Processes: A Concurrent Programming Concept," Communications of the ACM, Vol. 21, No. 11, November 1978, pp. 934 - 941.
50. Harrold, J.D. McGregor, and K.J. Fitzpatrick, "Incremental Testing of Object-Oriented Class Structures," Proceedings of the Fourteenth International Conference on Software Engineering, 1992, pp. 68 - 80.
51. Hathorn, "Structured Tasks," Proceedings of the Sixth Washington Ada Symposium, Association for Computing Machinery, New York, New York, 1989, pp. 265 - 272.
52. Hayes, Editor, Specification Case Studies, Prentice Hall, London, United Kingdom, 1987.
53. Hembold and D. Luckham, "Debugging Ada Tasking Programs," IEEE Software, Vol. 2, No. 2, March 1985, pp. 47 - 57.
54. Hembold and D. Luckham, "TSL: Task Sequencing Language," Ada in Use, Cambridge University Press, Cambridge, U.K., 1985.
55. Hetzel, The Complete Guide to Software Testing, Second Edition, QED Information Sciences, Inc. Wellesley, Massachusetts, 1988.
56. Hoare, "An Axiomatic Basis for Computer Programming," Communications of the ACM, Vol. 12, No. 10, October 1969, pp. 576 - 580, 583.
57. Hoare, "Proof of a Program: FIND," Communications of the ACM, Vol. 14, No. 1, January 1971, pp. 39 - 45.

58. Hoare, "Proof of Correctness of Data Representation," *Acta Informatica*, Vol. 1, 1972, pp. 271 - 181.
59. Hoare, "Communicating Sequential Processes," *Communications of the ACM*, Vol. 21, No. 8, August 1978, pp. 666 - 677.
60. Hoare, *Communicating Sequential Processes*, Prentice Hall, Englewood Cliffs, New Jersey, 1985.
61. Hobin, "The Ada Test and Verification System (ATVS)," *Conference Proceedings Tri-Ada '88*, 1988.
62. Hobin, "The Ada Test and Verification System (ATVS): An Overview," *Conference Proceedings Ada Expo '88*, 1988.
63. Hollocker, *Software Reviews And Audits Handbook*, John Wiley & Sons, New York, New York, 1990.
64. Honda and A. Yonezawa, "Debugging Concurrent Systems Based on Object Groups," in *ECOOP '88: Proceedings of the European Conference on Object-Oriented Programming*, Lecture Note on Computer Science, Volume 322, S. Gjessing and K. Nygaard, Editors, Springer Verlag, New York, New York, 1988, pp. 267 - 282.
65. Howden, *Software Engineering and Technology: Functional Program Testing*, McGraw-Hill, New York, New York, 1987.
66. Hseush and G.E. Kaiser, "Data Path Debugging: Data-Oriented Debugging for a Concurrent Programming Language," *Proceedings of the ACM SIGPLAN/SIGOPS Workshop on Parallel and Distributed Debugging*, Madison, Wisconsin, May 1988, pp. 236 - 246.
67. Humphrey, *Managing the Software Process*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.
68. Institute for Electrical and Electronics Engineers, *IEEE Standard ATLAS Test Language*, IEEE Standard 416-1981, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1981.
69. Institute for Electrical and Electronics Engineers, *An American National Standard: IEEE Standard for Software Test Documentation*, ANSI/IEEE Std 829-1983, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1983.
70. Institute for Electrical and Electronics Engineers, *An American National Standard: IEEE Standard for Software Verification and Validation Plans*, ANSI/IEEE Std 1012-1986, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1986.
71. Institute for Electrical and Electronics Engineers, *An American National Standard: IEEE Standard for Software Unit Testing*, ANSI/IEEE Std 1008-1987, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1986.
72. Institute for Electrical and Electronics Engineers, *An American National Standard: IEEE Standard for Software Reviews and Audits*, ANSI/IEEE Std 1028-1988, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1988.

73. Institute for Electrical and Electronics Engineers, Proceedings of the Second Workshop on Software Testing, Verification, and Analysis, 19-21 July 1988, Banff, Canada, 1988.
74. Institute for Electrical and Electronics Engineers, Software Engineering Standards, Third Edition, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1989.
75. Institute for Electrical and Electronics Engineers, IEEE Standard Glossary of Software Engineering Terminology, IEEE Standard 610.12-1990, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1991.
76. Institute for Electrical and Electronics Engineers, IEEE Standards Collection: Software Engineering, 1993 Edition, The Institute of Electrical and Electronics Engineers, Inc., New York, New York, 1993.
77. Isoda, "Visual Debugger VIPS: Visual Representation of Program Execution," Conference Proceedings Software Testing and Validation September 23-24, 1987, National Institute for Software Quality and Productivity, Inc., 1987, pp. E-1 - E-27.
78. Jalote, "Testing the Completeness of Specifications," IEEE Transactions on Software Engineering, Vol. 15, No. 5, May 1989, pp. 526 - 531.
79. Jones, Systematic Software Development Using VDM, Prentice Hall, Englewood Cliffs, New Jersey, 1986.
80. Jorgensen and C. Erickson, "Object-Oriented Integration Testing," Communications of the ACM, Vol. 37, No. 9, September 1994, pp. 30 - 38.
81. Juran, "Basic Concepts," in Quality Control Handbook, Third Edition, Edited by J.M. Juran, F.M. Gryna, Jr., and F.M. Bingham, Jr., McGraw-Hill, New York, New York, 1979.
82. Kaiser, D.E. Perry, and W.M. Schell, "INFUSE: Fusing Integration Test Management With Change Management," Proceeding of COMPSAC '89 -- The 13th Annual International Computer Software and Application Conference, Orlando, Florida, September 1989, pp. 552 - 558.
83. Kaner, Testing Computer Software, Tab Books, Blue Ridge Summit, Pennsylvania, 1988.
84. Knuth, The Art of Computer Programming, Volume 1: Fundamental Algorithms, Second Edition, Addison-Wesley Publishing Company, Reading, Massachusetts, 1973.
85. Laski, "Testing in the Program Development Cycle," Software Engineering Journal, Vol. 4, No. 2, March 1989, pp. 95 - 106.
86. Lazzerini and L. Lapriore, "Abstraction Mechanisms for Event Control in Program Debugging," IEEE Transactions on Software Engineering, Vol. 15, No. 7, July 1989, pp. 890 - 901.
87. Leveson, "Software Safety: What, Why, and How," ACM Computing Surveys, Vol. 18, No. 2, June 1986, pp. 125 - 164.
88. Lewis, Independent Verification and Validation: A Life-Cycle Engineering Process for Quality Software, John Wiley & Sons, New York, New York, 1992.

89. Lin and R. LeBlanc, "Event Based Debugging of Object/Action Programs, SIGPLAN Notices, Vol. 24, No. 1, January 1989, pp. 23 - 34.
90. Long and L.A. Clark, "Task Interaction Graphs for Concurrent Analysis," Proceedings of the Second Workshop on Software Testing, Verification, and Analysis, 19 - 21 July 1988, Banff, Canada, IEEE Catalog Number 88TH0225-3, 1988, pp. 134 - 135.
91. Luckham, R. Neff, and D. Rosenblum, "An Environment for Ada Software Development Based on Formal Specification," Ada Letters, Vol. 7, No. 3, 1987.
92. Mackay and G. Davenport, "Virtual Video Editing in Interactive Multimedia Applications," Communications of the ACM, Vol. 32, No. 7, July 1989, pp. 802 - 810.
93. Marks, Testing Very Big Systems, McGraw-Hill, Inc., New York, New York, 1992.
94. Martin, "Assurance Technology," Conference Proceedings Tri-Ada '88, 1988.
95. McCabe, "A Complexity Measure" IEEE Transactions on Software Engineering, December 1976, pp. 308 - 320.
96. McCabe, Editor, IEEE Tutorial: Structured Testing, IEEE Computer Society Press, Silver Spring, Maryland, 1982.
97. McCabe, "Automating the Testing Process Through Complexity Metrics," Conference Proceedings Software Testing and Validation September 23-24, 1987, National Institute for Software Quality and Productivity, Inc., 1987, pp. G-1 - G-30.
98. McCracken, "Testing Reusable Ada Software," Conference Proceedings Software Testing and Validation September 23-24, 1987, National Institute for Software Quality and Productivity, Inc., 1987, pp. P- 1 - P-27.
99. McGregor and T.D. Korson, Testing the Polymorphic Interactions of Classes, Technical Report No. TR-94-103, Clemson University, 1994.
100. McGregor and T.D. Korson, "Integrated Object-Oriented Testing and Development Processes," Communications of the ACM, Vol. 37, No. 9, September 1994, pp. 59 - 77.
101. Miller, and W.E. Howden, Tutorial: Software Testing and Validation Techniques, IEEE Computer Society Press, Washington, D.C., IEEE Catalog No. EHO180-0, 1981.
102. Mizuno, "Software Quality Improvement," IEEE Computer, Vol. 16, No. 3, March 1983, pp. 66 - 72.
103. Murphy, P. Townsend, and P.S. Wong, "Experiences With Cluster and Class Testing," Communications of the ACM, Vol. 37, No. 9, September 1994, pp. 39 - 47.
104. Musa and A.F. Ackerman, "Quantifying Software Validation: When to Stop Testing?," IEEE Software, Vol. 6, No. 3, May 1989, pp. 19 - 27.
105. Musa, A. Iannino, and K. Okumoto, Software Reliability: Professional Edition, McGraw-Hill Publishing Company, New York, New York, 1990.

106. Myers, Software Reliability Principles and Practices, John Wiley & Sons, New York, New York, 1976.
107. Myers, "A Controlled Experiment In Program Testing and Code Walkthroughs/Inspections," Communications of the ACM, Vol. 21, No. 9, September 1978, pp. 760 - 768.
108. Myers, The Art of Software Testing, John Wiley & Sons, New York, New York, 1979.
109. Neumann, "Flaws in Specifications and What to Do About Them," Proceedings of the Fifth International Workshop on Software Specification and Design, May 19-20, 1989, Pittsburgh, Pennsylvania, Published as Software Engineering Notes, Vol. 14 No. 3, IEEE Computer Society Press, Washington, D.C., May 1989, pp. xi - xv.
110. Olthoff, "Augmentation of Object-Oriented Programming by Concepts of Abstract Data Type Theory: The ModPascal Experience," OOPSLA '86 Conference Proceedings, special issue of SIGPLAN Notices, Vol. 21, No. 11, November 1986, pp. 429 - 443.
111. Ostrand, R. Sigal, and E. Weyuker, "Design for a Tool to Manage Specification Based Testing," Workshop on Software Testing, Banff, Canada, July 1987, pp. 41 - 50.
112. Ould and C. Unwin, Testing In Software Development, Cambridge University Press, Cambridge, United Kingdom, 1986.
113. Parrington and M. Roper, Understanding Software Testing, Halstead Press, New York, New York, 1989.
114. Parrish and S.H. Zweben, "Analysis and Refinement of Software Test Data Accuracy Properties," IEEE Transactions on Software Engineering, Vol. 17, No. 6, June 1991, pp. 565 - 581.
115. Perry and G.E. Kaiser, "Adequate Testing and Object-Oriented Programming," Journal of Object-Oriented Programming, Vol. 2, No. 5, January/February 1990, pp. 13 - 19.
116. Perry, How to Test Software Packages, John Wiley & Sons, New York, New York, 1986.
117. Perry, A Structured Approach To Software Testing, QED Information Sciences, Inc., Wellesley, Massachusetts, 1988.
118. Podgurski and L. Clarke, "The Implications of Program Dependencies for Software Testing, Debugging, and Maintenance," Software Engineering Notes, Vol. 14, No. 8, December 1989, pp. 168 - 178.
119. Poston, "Automated Testing from Object Models," Communications of the ACM, Vol. 37, No. 9, September 1994, pp. 48 - 58.
120. Poutanen, "Two Portable Ada Testing Tools TBGEN and TCMON," Ada Components: Libraries and Tools -- Proceedings of the Ada-Europe International Conference, Stockholm 26-28 May 1987, Cambridge University Press, Cambridge, United Kingdom, 1987, pp. 197 - 208.
121. Purchase and R.L. Winder, "Debugging Tools for Object-Oriented Programming," Journal of Object-Oriented Programming, Vol. 4, No. 3, June 1991, 10 - 14, 16 - 18, 20 - 24, 26 - 27.

122. Quirk, Editor, Verification and Validation of Real-Time Software, Springer-Verlag, New York, New York, 1985.
123. Radi, "TestGen--Testing Tool for Ada Designs and Ada Code," Proceedings of the Sixth National Conference on Ada Technology, 1988.
124. Radi, "Testing Ada Designs and Code--TestGen--Ada Testing Tool," Conference Proceedings Tri-Ada '88, ACM Press, New York, New York 1988, pp. 165 - 180.
125. Radi, "TestGen--Testing Tool for Ada Designs and Ada Code," Conference Proceedings Ada Expo '88, 1988.
126. Radi, "TestGen--Testing Tool for Ada Designs and Ada Code," Proceedings of the Sixth Washington Ada Symposium, Association for Computing Machinery, New York, New York, 1989, pp. 53 - 61.
127. Rapps and E.J. Weyuker, "Selecting Software Test Data Using Data Flow Information," IEEE Transactions on Software Engineering, Vol. 11, No. 4, April 1985, pp. 367 - 375.
128. Reifer, State of the Art in Software Quality Management, Reifer Consultants, Torrance, California, 1985.
129. Remkes, K. Gutzmann, and F. Sizer, "Automated Test Support for Ada PDL," Proceedings of the Sixth Washington Ada Symposium, Association for Computing Machinery, New York, New York, 1989, pp. 39 - 45.
130. Roe and J.H. Rowland, "Some Theory Concerning Certification of Mathematical Subroutines by Black Box Testing," IEEE Transactions on Software Engineering, Vol. SE-13, No. 7, July 1987, pp. 761 - 766.
131. Roetzheim, Developing Software to Government Standards, Prentice Hall, Englewood Cliffs, New Jersey, 1991.
132. Sankar, D. Rosenblum, and R. Neff, "An Implementation of Anna," Ada in Use, Cambridge University Press, Cambridge, U.K. 1985.
133. Schulmeyer and J.I. McManus, Editors, Handbook of Software Quality Assurance, Van Nostrand Reinhold, New York, New York, 1987.
134. Schulmeyer, Zero Defect Software, McGraw-Hill, Inc., New York, New York, 1990.
135. Shankar, "A Functional Approach to Module Verification," IEEE Transactions on Software Engineering, Vol. SE-8, No. 2, March 1982, pp. 147 - 160.
136. Shneiderman, Software Psychology, Winthrop Publishers, Inc., Cambridge, Massachusetts, 1980.
137. Smith and D.J. Robson, "Object-Oriented Programming: The Problem of Validation," Proceedings of the Sixth International IEEE Conference on Software Maintenance, November 1990, IEEE Computer Society Press, Los Alamitos, California, 1990, pp. 272 - 282.
138. Smith and D.J. Robson, "A Framework for Testing Object-Oriented Programs," Journal of Object-Oriented Programming, Vol. 5, No. 3, June 1992, pp. 45 - 53.

139. Sommerville, Software Engineering, Third Edition, Addison-Wesley Publishing Company, Reading, Massachusetts, 1989.
140. Spivey, Understanding Z: A Specification Language and Its Formal Semantics, Cambridge University Press, Cambridge, United Kingdom, 1988.
141. Spivey, The Z Notation: A Reference Manual, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
142. Strelch, "Ada Test and Verification System: ATVS," Proceedings of the First International Conference on Ada Programming Language Applications For the NASA Space Station, 1986.
143. Tai and C.Y. Din, "Validation of Concurrency in Software Specification and Design," Third International Workshop on Software Specification and Design, IEEE Computer Society Press, Silver Spring, Maryland, 1985, pp. 223 - 227.
144. Tai, R.H. Carver, and E.E. Obaid, "Deterministic Execution Debugging of Concurrent Ada Programs," Proceedings of Compsac '89, IEEE Computer Society Press, Silver Spring, Maryland, October, 1989, pp. 102 - 109.
145. Tai, R.H. Carver, and E.E. Obaid, "Debugging Concurrent Ada Programs by Deterministic Execution," IEEE Transactions on Software Engineering, Vol. 17, No. 1, January 1991, pp. 45 - 63.
146. Tai, "Reproducible Testing of Concurrent Ada Programs," Proceedings of SOFTFAIR II, IEEE Computer Society Press, Silver Spring, Maryland, December, 1985, pp. 114 - 120.
147. Tai, "On Testing Concurrent Programs," Proceedings of COMPSAC '85, IEEE Computer Society Press, Silver Spring, Maryland, October, 1985, pp. 310 - 317.
148. Tai, "A Graphical Representation of Rendezvous Sequences of Concurrent Ada Programs," Ada Letters, Vol. 6, No. 1, January/February 1986, pp. 94 - 103.
149. Tai, "Reproducible Testing of Ada Tasking Programs," Proceedings of the IEEE Second International Conference on Ada Applications and Environments, IEEE Computer Society Press, Silver Spring, Maryland, 1986, pp. 69 - 79.
150. Tai, "A Methodology for Testing Concurrent Ada Programs," Proceedings of The Joint Ada Conference Fifth National Conference on Ada Technology and Washington Ada Symposium, March, 1987, pp. 459 - 464.
151. Tassel, Program Style, Design, Efficiency, Debugging, and Testing, Second Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1978.
152. Tauson-Conte, "Ada Complexity Extension (ACE) An extension of McCabe's Cyclomatic Complexity Metric for Analysis of Ada Software," Proceedings of the Sixth National Conference on Ada Technology, 1988, pp. 7 - 12.
153. Taylor and C.D. Kelly, "Structural Testing of Concurrent Programs," Proceedings of the Workshop on Software Testing, Banff, Alberta, Canada, July 1986, pp. 164 - 169.

154. Taylor and D.L. Levine, "Structural Testing of Concurrent Programs," IEEE Transactions on Software Engineering, Vol. 18, No. 3, March 1992, pp. 206 - 215.
155. Taylor and L.J. Osterweil, "Anomaly Detection in Concurrent Software by Static Data Flow Analysis," IEEE Transactions on Software Engineering, Vol. SE-6, No. 3, May 1980, pp. 265 - 278.
156. Taylor and T.A. Standish, "Steps to an Advanced Ada Programming Environment," IEEE Transactions on Software Engineering, Vol. SE-11, No. 3, March 1985, pp. 302 - 309.
157. Taylor, "A General-Purpose Algorithm for Analyzing Concurrent Programs," Communications of the ACM, Vol. 26 No. 5, May, 1983, pp 362 - 376.
158. Trausan-Matu, J. Tepandi, and M. Barbuseanu, "Validation, Verification, and Testing of Object-Oriented Programs," Proceedings of the EastEurOOPe '91 Conference, September 15-19, 1991, Bratislava, Czechoslovakia, pp. 62 - 71.
159. von Henke, D. Luckham, B. Krieg-Brueckner, and O. Owe, "Semantic Specification of Ada Packages," Ada in Use, Cambridge Press, Cambridge, UK, 1985.
160. von Mayrhauser, "Testing and Evolutionary Development," Software Engineering Notes, Vol. 16, No. 4, October 1991, pp. 31 - 36.
161. Wallace and R. Fujii, "Verification and Validation: Techniques to Assure Reliability," IEEE Software, Vol. 6, No. 3, May 1989, pp. 8 - 9.
162. Wallace and R. Fujii, "Software Verification and Validation: An Overview," IEEE Software, Vol. 6, No. 3, May 1989, pp. 10 - 18.
163. Weinberg, The Psychology of Computer Programming, Van Nostrand Reinhold Company, New York, New York, 1971.
164. Weinberg, "Kill That Code," Infosystems, August 1983, p. 49.
165. Weiss, "A Formal Framework for the Study of Concurrent Program Testing," Proceedings of the Second Workshop on Software Testing, Verification, and Analysis, 19 - 21 July 1988, Banff, Canada, IEEE Catalog Number 88TH0225-3, 1988, pp. 106 - 113.
166. Weyuker and B. Jeng, "Analyzing Partition Testing Strategies," IEEE Transactions on Software Engineering, Vol. 17, No. 7, July 1991, pp. 703 - 711.
167. Weyuker, "Axiomatizing Software Test Data Accuracy," IEEE Transactions on Software Engineering, Vol. SE-12, No. 12, December 1986, pp. 1128 - 1138.
168. Weyuker, "The Evaluation of Program-Based Software Test Data Accuracy Criteria," Communications of the ACM, Vol. 31, No. 6, June 1988, pp. 668 - 675.
169. Weyuker, "The Cost of Data Flow Testing: An Empirical Study," IEEE Transactions on Software Engineering, Vol. 16, No. 2, February 1990, pp. 121 - 128.
170. Yonezawa and M. Tokoro, Editors, Object-Oriented Concurrent Programming, The MIT Press, Cambridge, Massachusetts, 1987.

171. Young and R.N. Taylor, "Combining Static Concurrency Analysis with Symbolic Execution," Proceedings of the Workshop on Software Testing, IEEE Computer Society Press, Washington, D.C., 1986, pp. 170 - 178.
172. Young, R.N. Taylor, K. Forester, and D. Brodbeck, "Integrated Concurrency Analysis in a Software Development Environment," Proceedings of the Third Workshop on Software Testing, Analysis, and Verification, Key West, Florida, December 1989, pp. 200 - 209.
173. Yourdon and M. Constantine, Structured Design, Prentice Hall, Englewood Cliffs, New Jersey, 1979.
174. Yourdon, Techniques of Program Structure and Design, Prentice Hall, Englewood Cliffs, New Jersey, 1975.
175. Yourdon, Managing The Structured Techniques, Fourth Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
176. Yourdon, Structured Walkthroughs, Fourth Edition, Prentice Hall, Englewood Cliffs, New Jersey, 1989.
177. Zweben and J. Gourlay, "On the Adequacy of Weyuker's Test Data Adequacy Axioms," IEEE Transactions on Software Engineering, Vol. 15, No. 4, April 1989, pp. 496 - 500.

TEST COVERAGE

1. Chusho, T.: Test Data Selection and Quality Estimation Based on the Concept of Essential Branches for Path Testing. IEEE Transactions on Software Engineering, 13(1987)5, pp. 509-517.
2. Emerson, T.J.: Program Testing, Path Coverage, and the Cohesion Metric. IEEE COMPSAC, 1984, pp. 421-431.
3. Horgan, J.R.; London, S.; Lyu, M.R.: Achieving Software Quality with Testing Coverage Measures. IEEE Computer, September 1994, pp. 60-68.
4. Lasalle, H.: A Structural Approach to Improvement the Testing Process. in Kelly, M.: Management and Measurement of Software Quality, UNICOM SEMINARS, Middlesex, UK, 1993, pp. 121-130.
5. Malevris, N.; Yates, D.F.; Veevers A.: Predictive metric for likely feasibility of program paths. Information and Software Technology, 32(1990)2, pp. 115-118.
6. McAllister, M.; Vuong, S.T.; Alilivic-Curgus, J.: Automated Test Case Selection Based on Test Coverage Metrics. Proceedings of the INPTS`92, Montreal, Sept/Oct 1992, pp. 63-76.
7. McColl, R.B.; McKim, J.C.: Evaluating and Extending NPath as a Software Complexity Measure. The Journal of Systems and Software, 7(1992) pp. 275-279.
8. Nejme, B.A.: NPATh: A Measure of Execution Path Complexity and Its Applications. Comm. of the ACM, 31(1988)2, pp. 188-200.
9. Piwowarski, P.; Ohba, M.; Caruso, J.: Coverage Measurement Experience During Function Test. Proceedings of the 15th International Conference on Software Engineering, May 17-21, Baltimore, 1993, pp. 287-301.

10. Spilner, A.: Test Coverage Metrics for the Integrations Testing. (German) in: Dumke/Zuse: Theorie und Praxis der Softwaremessung, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp. 59-77.
11. Staknis, M.E.: Software quality assurance through prototyping and automated testing. Information and Software Technology, 32(1990)1, pp. 26-33.
12. Tai, K.: Predicate-Based Test Generation for Computer Programs. Proceedings of the 15th International Conference on Software Engineering, May 17-21, Baltimore, 1993, pp. 267-276.
13. Woodward, M.R.; Hedley, M.A.; Hennell, M.A.: Experience with Path Analysis and Testing of Programs. IEEE Transactions on Software Engineering, 6(1980)3, pp. 278-286.

TEST METRICS

1. Bache, R.; Muellerburg, M.: Measures of testability as a basis for quality assurance. Software Engineering Journal, March 1990, pp. 85-92.
2. Bainbridge, J.: Defining Testability Metrics Axiomatically. Software Testing, Verification and Reliability, 4(1994), pp. 63-80.
3. Basili, V.R.; Selby, R.W.: Comparing the Effectiveness of Software Testing Strategies. IEEE Transactions on Software Engineering, 13(1987)12, pp. 1278-1296.
4. Bastani, F.B.; DiMarco, G.; Pasquini, A.: Experimental Evaluation of a Fuzzy-Set Based Measure of Software Correctness Using Program Mutation. Proceedings of the 15th International Conference on Software Engineering, May 17-21, Baltimore, 1993, pp. 45-54.
5. Bochmann; G.v. et al.: Summary of Discussion on 'Design for Testatbility'. Montreal, July, 1992.
6. Briand, L.C.; Basili, V.R.; Hetmanski, C.J.: Providing an Empirical Basis for Optimizing the Verification and Testing Phases of Software Development. Proceedings of the Third International Symposium on Software Reliability Engineering, Research Triangle Park, NC, October 8-9, 1992, pp. 329-338.
7. Carr, G.: Independent Testing at the Prudential. First European International Conference on Software Testing, Analysis & Review (EuroStar), London, October 25-28, 1993, pp. 37-42.
8. Davey, S. et. al.: Metrics Collection in Code and Unit Test as Part of Continous Quality Improvement. First European International Conference on Software Testing, Analysis & Review (EuroStar), London, October 25-28, 1993, pp. 37-42.
9. Ebert, Ch.; Liedtke, T.: Experience in complexity-based error estimation. (german) in: Dumke/Zuse: Theorie und Praxis der Softwaremessung, Deutscher Universitaetsverlag, Wiesbaden, 1994, pp.78-96.
10. Evanco, W.M.: Ordered Response Models for the Analysis of Software Fault Correction Effort. Proceedings of the Annual Oregon Workshop on Software Metrics, April 10-12, 1994, Silver Falls, Oregon.

11. Harrison, W.: Using Software Metrics to Allocate Testing Resources. *Journal of Management Information Systems*, 4(1988)4, pp.~93-105.
12. IEEE Standard for Software Test Documentation. IEEE Std 829-1983, New York, February 1983.
13. Katkov, V.L.; Shimarov, V.A.: A Quantitative Approach to Program Testing Quality Control. *Proceedings of the Second International Conference on Software Quality*, Research Triangle Park, NC, October 3-5, 1992, pp. 215-222.
14. Katkov, V.L.; Shimarov, V.A.: Structural Software Testing Criteria Evaluation. *Proceedings of the International Conference on CAD/CAM, Robotics and Practice of the Future*, St. Petersburg, 1993, pp. 558-565.
15. Koshgoftaar, T.M.; Szabo, R.M.: ARIMA Models of Software System Quality. *Proceedings of the Annual Oregon Workshop on Software Metrics*, April 10-12, 1994, Silver Falls, Oregon.
16. Muellerburg, M.: *Fundamental Concepts of Software Testing*. GMD, Birlinghoven, March 1991.
17. Muellerburg, M.: *Software Testing: A Stepwise Process*. *Proceedings of the Second European Conference on Software Quality Assurance*, Oslo, 1990.
18. Royer, T.C.: *Software Testing Management - Life on the Critical Path*. Prentice-Hall Inc., 1993.
19. Schaefer, H.: *Inspection Handbook for Computer Program Development Projects*. Report No. 84 01 41-9, Oslo, July 1984.
20. Schaefer, H.: *Organization and managing of the software testing (German)*. Tutorium-Heft, Schweizerische Arbeitsgemeinschaft fuer Qualitaetsfoerderung, Zuerich, September 1991.
21. Seddio, C.: Integrating Test Metrics within a Software Engineering Measurement Program at Eastman Kodak Company: A Follow-up Case Study. *The Journal of Systems and Software*, 20(1993)3, pp. 227-235.
22. Shimarov, V.A.: Definition and quantitative evaluation of test criteria. *Proc. of the Fourth European Conference on Software Quality*, October 17-20, Basel, Switzerland, pp. 350-360.
23. Voas, J.M.; Miller, K.W.: Semantic Metrics for Software Testability. *The Journal of Systems and Software*, 20(1993)3, pp. 207-216.
24. Weyuker, E.J.: Can We Measure Software Testing Effectiveness? *Proceedings of the First International Software Metrics Symposium*, Baltimore, May 21-22, 1993, pp. 100-107.
25. Zhu, H.; Hall, P.A.V.: Test data adequacy measurement. *Software Engineering Journal*, January 1993, pp. 21-29. see in *Code Metrics: McCabes Cyclomatic Number*

TEXT ANALYSIS

1. Cherry, L.: Computer Aids for Writers. SIGPLAN Notices, 16(1981)6, pp. 61-67 Lehner, F.: Measurement of the quality of software documentations (german). WHU Research Paper, Nr. 10, Koblenz, August 1992.

VISUAL PROGRAMMING

1. Garcia, M.; Badre, A.N.; Stasko, J.T.: Development validation of icons varying in their abstractness. Interacting with Computers, 6(1994)2, pp. 191-211.
2. Glinert, E. P.: Towards software metrics for visual programming. International Journal of Man-Machine Studies, 30(1989), pp. 425-445.